NAVAL
POSTGRADUATE
SCHOOL

MONTEREY, CALIFORNIA

THESIS

MODELING AND SIMULATION OF SYNTHETIC APERTURE RADARS IN MATLAB

by

Brandon J. Fason

June 2013

Thesis Advisor:	Frank Kragh
Second Reader:	R. Clark Robertson

Approved for public release; distribution is unlimited

THIS PAGE INTENTIONALLY LEFT BLANK

REPORT DOCUMENTATION PAGE	*Form Approved OMB No. 0704-0188*

1. AGENCY USE ONLY *(Leave blank)*	**2. REPORT DATE** June 2013	**3. REPORT TYPE AND DATES COVERED** Master's Thesis

4. TITLE AND SUBTITLE MODELING AND SIMULATION OF SYNTHETIC APERTURE RADARS IN MATLAB	**5. FUNDING NUMBERS**
6. AUTHOR(S) Brandon J. Fason	
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) Naval Postgraduate School Monterey, CA 93943-5000	**8. PERFORMING ORGANIZATION**
9. SPONSORING /MONITORING AGENCY NAME(S) AND ADDRESS(ES) N/A	**10. SPONSORING/MONITORING AGENCY REPORT NUMBER**

11. SUPPLEMENTARY NOTES The views expressed in this thesis are those of the author and do not reflect the official policy or position of the Department of Defense or the U.S. Government. IRB Protocol number: N/A.

12a. DISTRIBUTION / AVAILABILITY STATEMENT **Approved for public release; distribution is unlimited**	**12b. DISTRIBUTION CODE** A

13. ABSTRACT (maximum 200 words)

The goal of this research is to produce a synthetic aperture radar (SAR) simulation in MATLAB that will provide a framework to investigate the effects of signal type and emission parameters on the resolution and accuracy of the simulated image. A total of three simulations were created for this research. The first two, which are range imaging and cross-range imaging, are the fundamental components of SAR and were vital to understanding the complexities of SAR. The signal types to be studied are a sinusoidal rectangular pulse, a linearly frequency modulated chirp, and band-limited Gaussian white noise. The simulations show that it is possible to produce usable images from each of the signal types examined, with the LFM chirp signal consistently producing the greatest resolution. Using noise as an input signal produced an exciting result with results almost on par with a sinusoidal pulse.

14. SUBJECT TERMS: Synthetic Aperture Radar, SAR, Range Imaging, Cross-Range Imaging, Digital Signal Processing, Match Filter	**15. NUMBER OF PAGES** 83
	16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT Unclassified	**18. SECURITY CLASSIFICATION OF THIS PAGE** Unclassified	**19. SECURITY CLASSIFICATION OF ABSTRACT** Unclassified	**20. LIMITATION OF ABSTRACT** UU

THIS PAGE INTENTIONALLY LEFT BLANK

MODELING AND SIMULATION OF SYNTHETIC APERTURE RADARS IN MATLAB

Brandon J. Fason
Lieutenant, United States Navy
B.S., United States Naval Academy, 2006

Submitted in partial fulfillment of the
requirements for the degree of

MASTER OF SCIENCE IN ELECTRICAL ENGINEERING

from the

NAVAL POSTGRADUATE SCHOOL
June 2013

Author: Brandon J. Fason

Approved by: Frank Kragh
Thesis Advisor

R. Clark Robertson
Second Reader

R. Clark Robertson
Chair, Department of Electrical and Computer Engineering

THIS PAGE INTENTIONALLY LEFT BLANK

ABSTRACT

The goal of this research is to produce a synthetic aperture radar (SAR) simulation in MATLAB that will provide a framework to investigate the effects of signal type and emission parameters on the resolution and accuracy of the simulated image. A total of three simulations were created for this research. The first two, which are range imaging and cross-range imaging, are the fundamental components of SAR and were vital to understanding the complexities of SAR. The signal types to be studied are a sinusoidal rectangular pulse, a linearly frequency modulated chirp, and band-limited Gaussian white noise. The simulations show that it is possible to produce usable images from each of the signal types examined, with the LFM chirp signal consistently producing the greatest resolution. Using noise as an input signal produced an exciting result with results almost on par with a sinusoidal pulse.

THIS PAGE INTENTIONALLY LEFT BLANK

TABLE OF CONTENTS

THIS PAGE INTENTIONALLY LEFT BLANK

LIST OF FIGURES

THIS PAGE INTENTIONALLY LEFT BLANK

EXECUTIVE SUMMARY

Synthetic aperture radar (SAR) is an important engineering advancement that has numerous applications throughout the military and civilian worlds. SAR is unique because it incorporates the motion of the platform into the digital signal processing in order to simulate an improbably large physical radar antenna. A transceiver is affixed to a plane or satellite and performs numerous transmit/receive cycles along the path of flight. The recorded reflected radio waves then undergo digital signal processing to generate an image of the target area.

The goal of this research is to generate a functional and easily understood SAR simulation in MATLAB and study the effects produced from varying the signal types used in transmission. The signal types studied are a sinusoidal pulse, a linear frequency modulated (LFM) chirp signal, and band-limited Gaussian white noise.

Synthetic aperture radar, in concept only, should be thought of as the combination of range and cross-range imaging. Range imaging, at its most basic, is transmitting a radio frequency (RF) pulse and recording the echoed signal return time and amplitude level. Simple calculus can then be used on the recorded signal returns to determine the distance of the target by dividing the round-trip time by the RF propagation speed. The simulation created for the range imaging section implements a more advanced version than the scenario above. The first improvement is a method for easily changing the transmitted signal type. This was accomplished by removing all of the signal characteristics from the main range imaging script and creating a function

that controls all of the input signal properties. Removing the signal properties from the main body of code increases readability and provides a process for easily modifying the signal characteristics. The second improvement was necessitated by the goal of varying the signal types, and involved implementing a matched filter into the signal processing. The LFM chirp signal produced the best resolution under the testing parameters. The rectangular sinusoidal pulse was second, and accurately ranged the targets and reflectivities but did not have the same quality of resolution found with the LFM chirp. The final signal type tested in the range imaging simulation was band-limited Gaussian white noise. The white noise produced interesting results by accurately ranging the target scene; however, it was not able to correctly return the reflectivities of the targets.

The next component of SAR is the cross-range imaging problem. In this scenario the target scene is fixed with all targets placed in a line parallel to the path of movement. The path of movement is known as the synthetic aperture, or slow-time domain. It is called the slow-time domain because of the large disparity between the speed of the RF wave propagation and that of the aircraft or satellite. It is due to this very large difference in speed that it was assumed the platform comes to a stop, undergoes a transmit/receive cycle then continues along the course of the aperture. The signal characteristics were removed from the main simulation for cross-range imaging as well, though it is less significant than it is in range imaging. The reason is that in cross-range imaging the signal type plays far less of a part than it does in range imaging. In cross-range imaging the most significant aspects are the aperture length, the carrier frequency, distance to the center of the target area, and the angle to the target area. This simulation makes the assumption that the target is always held to a broadside aspect to reduce coding complexity. The other three aspects are ways of synthetically or actually increasing the usable bandwidth, which allows greater imaging resolution. Increasing the resolution requires a higher carrier frequency, larger synthetic aperture, or reducing the distance to the center of the target area.

By combining the range imaging and cross-range imaging simulations a working SAR simulation was achieved. After establishing a baseline to work with utilizing the LFM chirp signal, the SAR simulation was subjected to numerous tests in order to understand the effects of each signal type. The LFM chirp performed the best again producing the greatest resolution of the three tested signal types. The rectangular sinusoidal pulse performed well, accurately locating and correctly depicting the targets in the scene. Finally the band-limited Gaussian noise pulse produced results that are almost on par with the sinusoidal pulse, which is consistent with the narrow cross-correlation for the noise pulse.

The simulation created for this thesis successfully modeled a synthetic aperture radar system and explored the effects of various signal types on the produced image.

THIS PAGE INTENTIONALLY LEFT BLANK

ACKNOWLEDGMENTS

I would like to acknowledge all the professors at the Naval Postgraduate School for their excellent teaching and guidance throughout my time. In particular, I would like to acknowledge my thesis advisor Professor Frank Kragh for his help, patience, and understanding during my thesis writing process.

Lastly, I would like to thank my family for being ever encouraging and supportive. It was their love and belief in me during the life changing events that occurred at the end of my thesis process that has given me the strength to finish.

THIS PAGE INTENTIONALLY LEFT BLANK

I. INTRODUCTION AND BACKGROUND

A. INTRODUCTION

A radar image is the aggregate of all the radar backscatters generated from "illuminating" the target area with radio frequency energy. Each pixel, or picture element, is represented by the intensity of the received return with darker areas being areas with low reflectivities, and correspondingly brighter areas having a higher reflectivity [1]. Illustrated in Figure 1 are the different types of radar images that are generated from a variety of scene types.

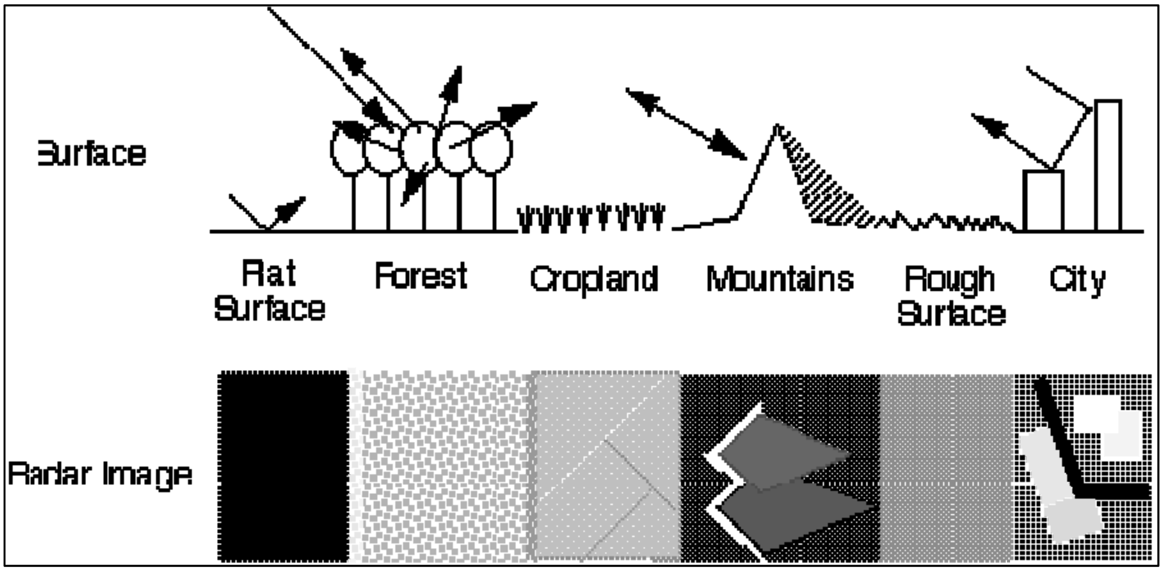

Figure 1. Imaging different types of surface with radar From [2].

Synthetic aperture radar (SAR) offers a means to generate high resolution images, from coherent digital processing of multiple radio frequency (RF) transmissions along a path of flight. Due to RF propagation characteristics a stationary antenna, with an appropriate size to fit onto an aircraft or satellite, would have great difficulty discerning targets on the ground because the radar footprint would be very large as can be seen from one transmit and receive position in Figure 2.

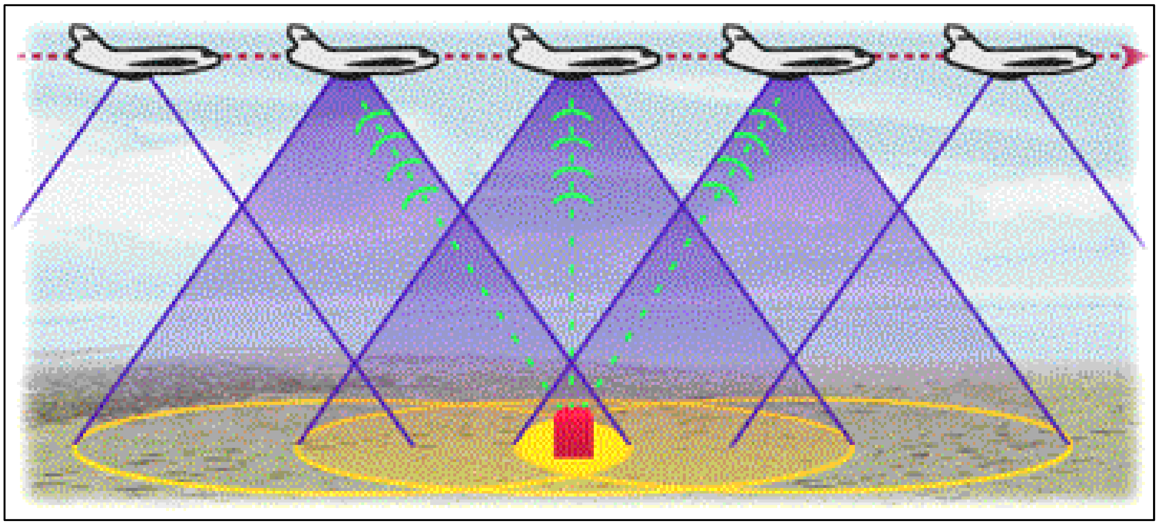

Figure 2. Multiple Radar Beams From [3].

In traditional radar, to achieve a higher range resolution, which is the distance between two targets that can reliably be seen as separate, a much larger antenna would be required in order to produce a narrower beam pattern. Synthetic aperture radar changes that by simulating a very large antenna that is physically improbable. It does this through digital signal processing on the recorded returns from numerous transmit receive cycles, all aimed at the same target area, along a path of flight similar to that which is represented in Figure 2.

The algorithm that will be presented utilizes matched filtering of the received signal along with a generated reference signal. This method is less sensitive to estimation errors than other SAR imaging schemes are, such as the Range Doppler Algorithm, since the matched filter maximizes the signal power to noise power ration.

The goal of this thesis is to two-fold. First, produce an easily understandable computer simulation that implements the aforementioned matched filtering algorithm for the range, cross-range, and synthetic aperture radar imaging problems. The second goal is to research the effects produced from varying the input signal type on the generated images.

B. BACKGROUND

It is important to have a proper understanding of the history of SAR and radar in general to aid in understanding the complexities that make SAR work. RADAR is an acronym coined by the United States Navy in 1940 standing for **ra**dio **d**etection **a**nd **r**anging. Initially developed to provide ships navigation situational awareness during periods of severe weather that otherwise reduced or negated effective visual navigation, radar then progressed to land and air-based usage. World War II produced very rapid advances in radar technology with all of the major powers involved pursuing it as a critical and necessary capability. The radar systems provided range by calculating the round trip time of a RF pulse to reach a target and reflect back and relative direction of a target by comparing the returned signal level to the direction the antenna transmitted. In 1951, Carl A. Wiley, a mathematician at Goodyear Aircraft

Company, is credited with developing the process of SAR through post processing of the Doppler shift information and utilizing wavefront reconstruction theory [4]. This was the origin of radar imaging, producing two-dimensional images of target areas constructed from radar alone. Radar imaging filled the capability gap that was left when optical sensors were nullified due to poor light or inclement weather conditions.

On June 28, 1978, NASA launched SEASAT, the first satellite put into space with SAR capabilities. The purpose of this satellite was remote sensing of the Earth's oceans.

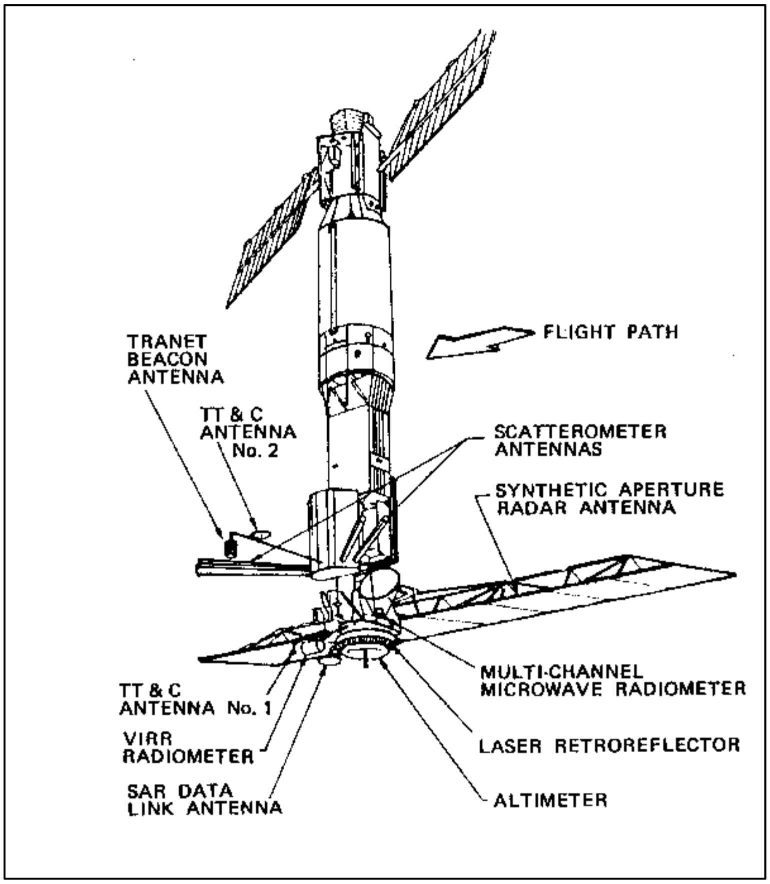

Figure 3. SEASAT From [5].

SEASAT, shown in Figure 3, operated on the L-band frequency of 1.275 GHz with a nearly circular orbit and altitude of 800 km. The system used a bandwidth of 19 MHz, an angle of incidence of 23 degrees, and a swath width of 100 km which resulted in a theoretical resolution on the surface of 25 square meters [5]. Prior to the SEASAT launch, SAR processing was done optically by utilizing Fourier optics methods including a coherent laser and a carefully positioned system of lenses to determine the Fourier transform of the data collected and stored on film. It was the limitations posed by optical processing that prompted research into digital processing, and the first SAR digital processor was launched as part of the SEASAT mission. When implemented in 1978, using a fast Fourier transform, the digital SAR processor was able to process a 40 square kilometer region with a resolution of 25 square meters in approximately 40 hours. Modern personal computers are capable of doing the same processing in a matter of milliseconds [1],[2].

THIS PAGE INTENTIONALLY LEFT BLANK

II. SYNTHETIC APERTURE RADAR

A. RANGE IMAGING

1. General Case

The general case of range imaging is easily relatable by most people. It is in its most simplistic terms a digitally processed version of echo location. An easy experiment, which most people have done as children, is to yell in a canyon and listen to the different returns of their voice. The staggered returns of the burst of sound energy are audibly demonstrating the differing ranges at which large objects are reflecting the sound waves. The reflecting objects need to be large relative to the wavelengths of audible tones and large enough to reflect sufficient acoustic energy. To the perceptive ear, differing levels of intensity and differing arrival times can be discerned, which are attributed to the reflectivity and the distance of the objects.

The canyon echo location experiment is the most basic version of modern day range imaging. To the aforementioned experiment digital signal processing is added, to create a ranging system. This ranging system is composed of a transceiver and signal processing system.

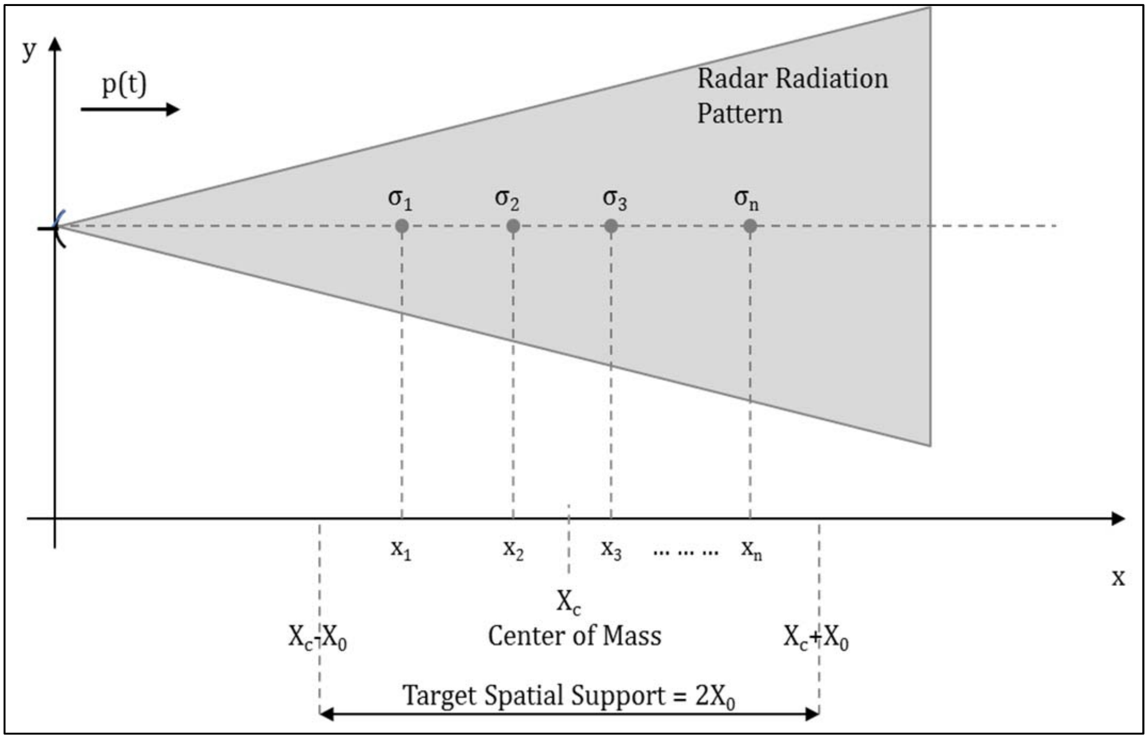

Figure 4. System Geometry for Range Imaging After [6].

Reference [6] gives the ideal target function, which represents the most coherent range-domain function that can represent the range and reflectivity of the targets in the x (range) domain:

$$f_0(x) = \sum_n \sigma_n \delta(x - x_n), \tag{1}$$

where σ_n is the reflectivity of the n^{th} target, $\delta(\;)$ is the Dirac delta function, and x_n is the distance in meters of the n^{th} target from the radar to the center of the target area. The results obtained using (1) are only simplifications of real world results, because targets are distributed and not located as lumps at points. Furthermore, it is impossible to produce the infinite bandwidth necessary in the illuminating signal for the resolution implied by the impulse function. If we illuminate the targets using a pulsed signal with a finite bandwidth, the received echoed signal is

$$\begin{aligned} s(t) &= p(t) * f_0\left(\tfrac{ct}{2}\right) \\ &= \sum_n \sigma_n p(t - t_n), \end{aligned} \tag{2}$$

where $t_n = \frac{2x_n}{c}$ is the round-trip delay for the radar signal to travel from the radar transceiver to the n^{th} target and back to the transceiver, c is the propagation speed, and * denotes time domain convolution[6].

The optimal reconstruction method is based on matched filtering. The time domain convolution, denoted by the first * in (3), is the operation that produces $s_M(t)$, where $s(t)$ is the received echoed signal, $p*(-t)$ is the matched filter impulse response, and the second * denotes complex conjugate.

$$s_M(t) = s(t) * p^*(-t). \tag{3}$$

It is the pulsed signal $p(t)$ that will be explored in greater detail in the following sections. We will look at the effects of varying the pulse duration T_p as well as differing signal types. The signal types to be investigated are a standard radio frequency (RF) sinusoidal pulse, a linearly frequency modulated pulse (LFM Chirp), and band-limited white Gaussian noise.

The simulation results that follow in sections 2 through 4 are the product of MATLAB code created specifically for this purpose. Although it is original in implementation, it draws heavily upon the *range.m* file supplied in [6]. One particular assumption of note that is made is that the range swath echo time period (T_x) is always greater than the pulse duration (T_p). This assumption removes the necessity of double coding for the case where $T_x < T_p$ which would introduce overlap between sent and received pulses, and is done for ease of implementation. The target definitions are set to be static for the purpose of comparison between the differing signal types, though it is necessary to note that the code does provide for random generation of number as well as location and reflectivity of targets. To support future development of the project, signal generation was removed from the main body of code and implemented as a separate function along with all necessary signal properties. In that way the signal generation is entirely independent of the remainder of the code which allows for easy and seamless changes between signal types.

2. Rectangular Sinusoidal Pulse

The range imaging simulation developed accepts a user's input that determines the signal type. In this section the rectangular sinusoidal pulse will be investigated.

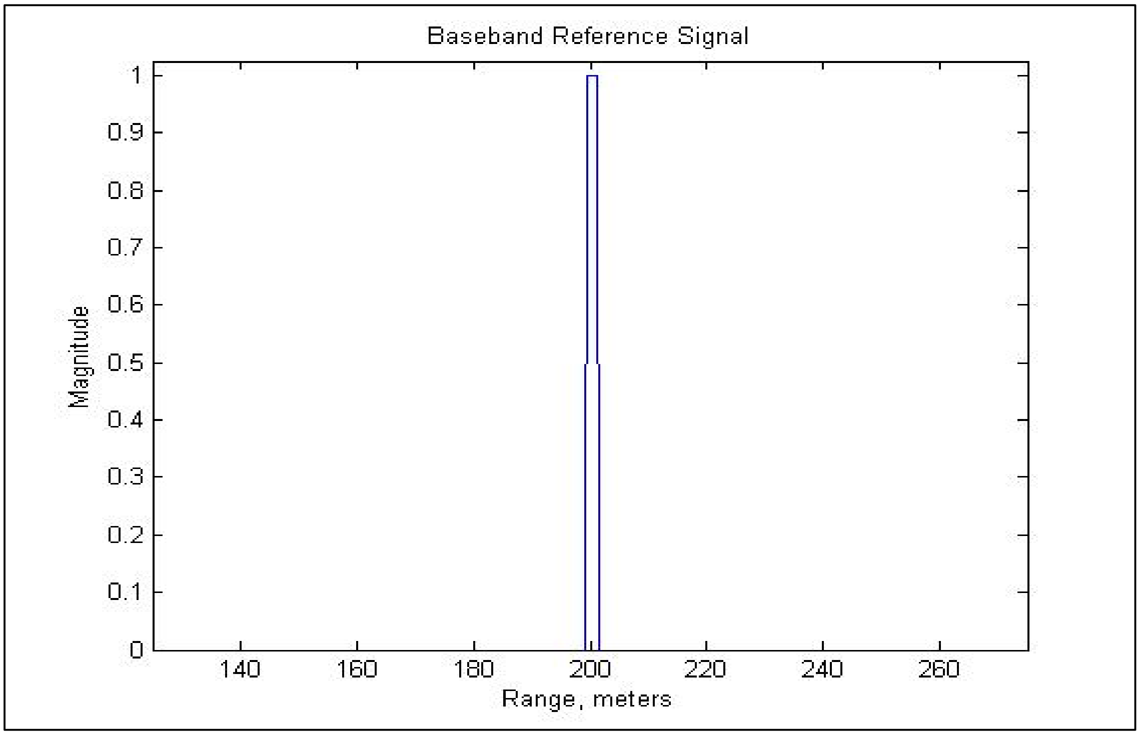

Figure 5. Rectangular Sinusoidal Pulse Baseband Reference Signal.

The simulation is based on a simple rectangular sinusoidal pulse of duration T_p,

$$p_{bb}(t) = I_{[0,T_p]}(t), \tag{4}$$

where the indicator function is defined as

$$I_A\left(t\right)=\begin{cases}1 & \text{if} \quad t\in A \\ 0 & \text{if} \quad t\notin A\end{cases}. \qquad (5)$$

This produces a baseband pulse p_{bb} of duration T_p similar to the baseband reference signal depicted in Figure 5 where the range is half the product of the speed of light and the propagation time. This baseband pulse is then upconverted and expressed as an analytic signal

$$p(t)=p_{bb}(t)e^{j\omega_c t}, \qquad (6)$$

where ω_c is the carrier frequency in radians per second. The signal in (6) is then emitted from the transmit antenna and reflected off the targets represented by (1), resulting in a received signal

$$\begin{aligned} s(t) &= p\left(t\right)* f_0\left(\tfrac{ct}{2}\right) \\ &= \sum_n \sigma_n p\left(t-t_n\right), \end{aligned} \qquad (7)$$

where

$$t_n=\tfrac{2x_n}{c}. \qquad (8)$$

The reflected signal is then returned to its baseband form

$$s_{bb}(t)=s(t)e^{-j\omega_c t}. \qquad (9)$$

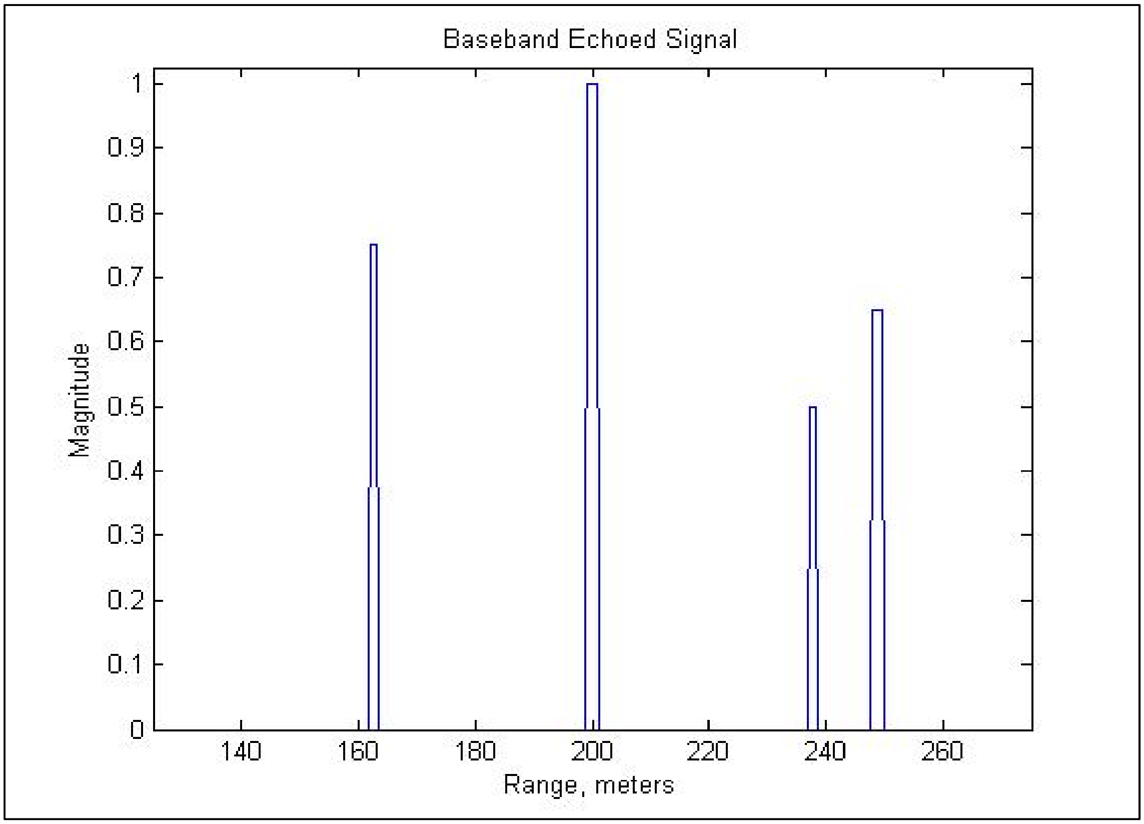

Figure 6. Rectangular Sinusoidal Pulse Baseband Echoed Signal.

The magnitude of s_{bb} is plotted against the range domain sample spacing which produces Figure 6, the baseband echoed signal. A matched filter is then used in conjunction with the baseband echoed signal to produce, $s_M(t)$, the range reconstructed target function via matched filtering as shown in Figure 7. The transition from a function of time to one of position is given by:

$$s_{bb}(t) = s_{bb}\left(\frac{2x}{c}\right), \tag{10}$$

The target area for these simulations was centered at X_c equal to 200 meters with width $2X_0$ equal to 150 meters, which defines the target spatial support region $(X_c - X_0, X_c + X_0)$.

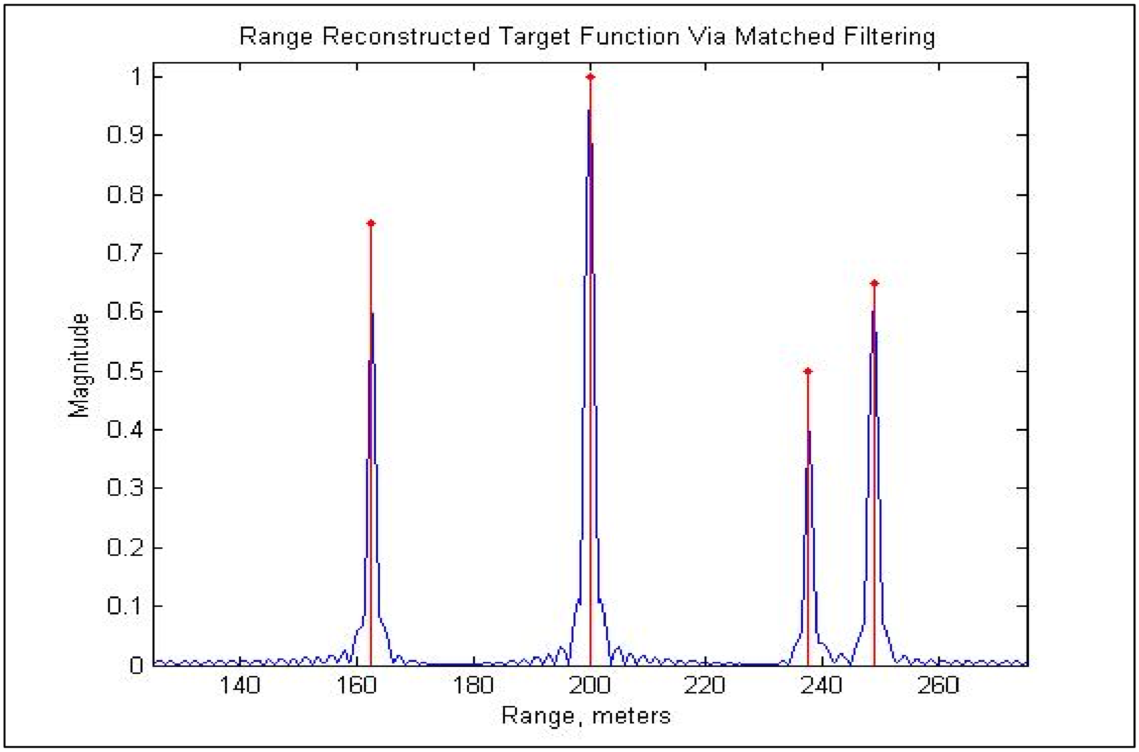

Figure 7. Rectangular Sinusoidal Pulse Range Reconstruction via Matched Filtering.

The magnitude of (10), $\left|s_M(t)\right|$ shown in Figure 7, is the reconstructed target function via matched filtering in blue and the ideal target function overlaid in red. The transmitted signal has a Fourier transform with a $\frac{\sin(x)}{x}$ shape with bandwidth and other parameters:

- Sinusoidal of the form: $e^{j2\pi f_c t} I_{[0,T_p]}(t)$
- Pulse duration: $T_p = 10\text{ns}$
- Signal Energy: 40nJ
- Carrier Frequency: $f_c = 1\text{GHz}$
- Null-to-Null Bandwidth: $\text{BW} = 200\text{MHz}$

3. Linearly Frequency Modulated Pulse (LFM Chirp)

The LFM chirp signal is generated similarly to the sinusoidal pulse. In the case of the rectangular pulse the indicator function set up the pulse itself, in the case of the LFM chirp it acts as a windowing function instead. The chirp signal used in the simulation is linearly increasing in frequency, and thus said to be an "up-chirp" whose analytic signal is given by

$$y(t) = e^{j\omega_{cm}t + j\alpha t^2} I_{[0,T_p]}(t), \tag{11}$$

where ω_{cm} is the modified carrier frequency

$$\omega_{cm} = \omega_c - \alpha T_p, \tag{12}$$

ω_c is the unmodified carrier frequency in radians per second, and α is the chirp rate in radians per second squared. The instantaneous frequency of the chirp signal is given by $\omega_{cm} + 2\alpha t$, which can be shown to sweep the range $\left(\omega_c - \alpha T_p, \omega_c + \alpha T_p\right)$ and is consistent with the baseband equivalent signal displayed in Figure 8.

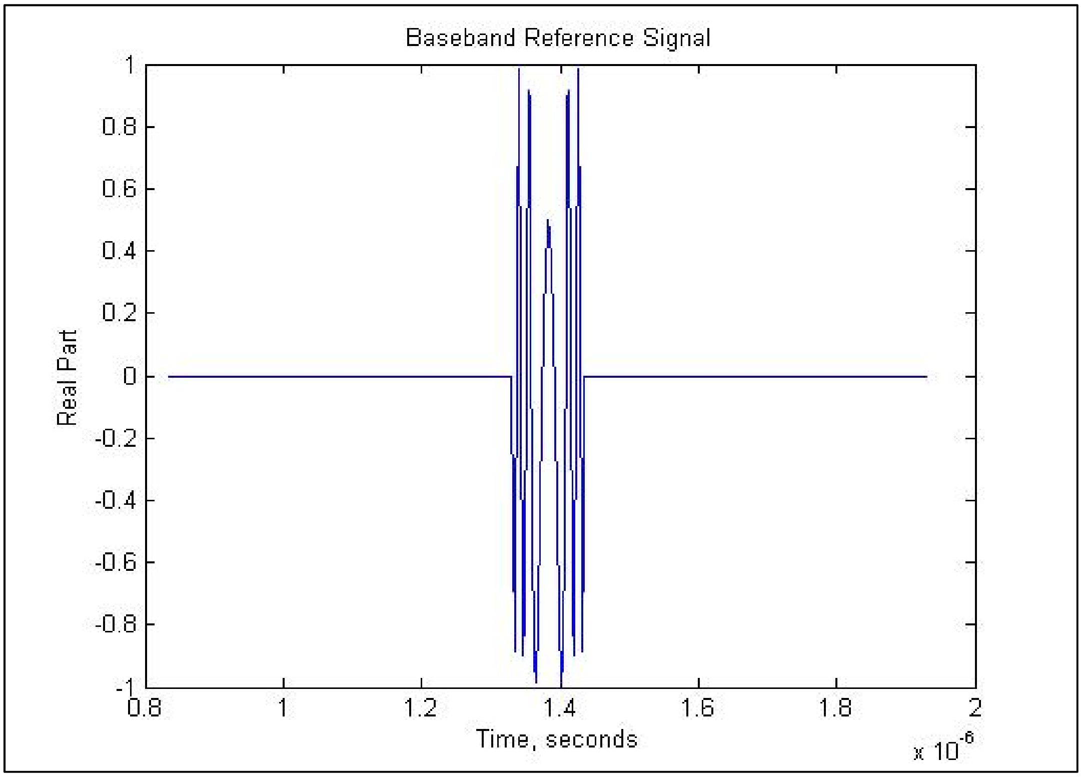

Figure 8. LFM Chirp Baseband Reference Signal.

One property of a chirp signal to note is that for the same pulse duration as a rectangular sinusoidal pulse there is a greater resolution achieved with a chirp signal. Target resolution is a measure of how closely targets can be placed and still recognized as unique. Increasing the bandwidth of the transmitted signal improves the target resolution.

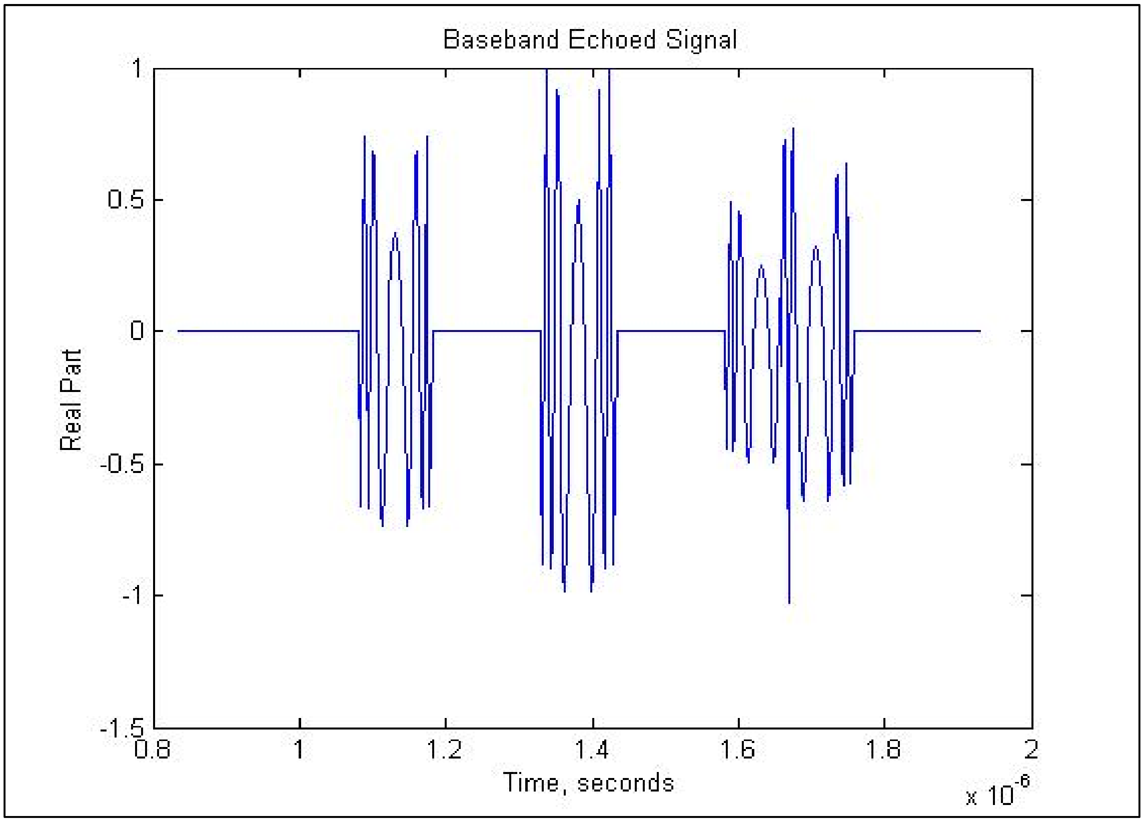

Figure 9. LFM Chirp Baseband Echoed Signal.

The same assumption of $T_x < T_p$ is made in the case of the chirp for the same reasons as stated previously. The simulation first calculates the time delay of the signals round-trip propagation for each target and uses this along with the given target reflectivity to create the received echoed signal that, after conversion to baseband, is represented in Figure 9. The baseband reference signal used as the matched filter impulse response is calculated and illustrated in Figure 8. The echoed signal goes through the same process described in the preceding section. First it is returned to its baseband components and transformed into the frequency domain by a Fourier transform, implemented in MATLAB by the fast Fourier transform *fft.m* routine. Once in the frequency domain it is element-wise multiplied by the complex conjugate of the reference signal, also in the frequency domain. The result is then transformed back into the time domain via an inverse Fourier transform implemented in MATLAB by the *ifft.m* routine. This solution is equivalent to convolution in the time domain and computationally less intensive. Illustrated in Figure 10 is the scaled magnitude of the matched filter output for the LFM chirp signal. The same target scene composition as the rectangular pulse was utilized.

As before, the blue curve is the magnitude of the reconstructed target function obtained by matched filtering and the red stem plot overlaid is the ideal target function. For this signal type the bandwidth has been calculated from the range of instantaneous frequencies. The signal parameters for this section of the simulation are as follows:

- LFM Chirp given by (11)
- Pulse duration: $T_p = 10\text{ns}$
- Signal Energy: 40nJ
- Carrier Frequency: $f_c = 1\text{GHz}$
- Chirp rate (α): $2\pi \times 10^{16} \text{rad/s}^2$
- Null-to-Null Bandwidth: $\text{BW} = 200\text{MHz}$

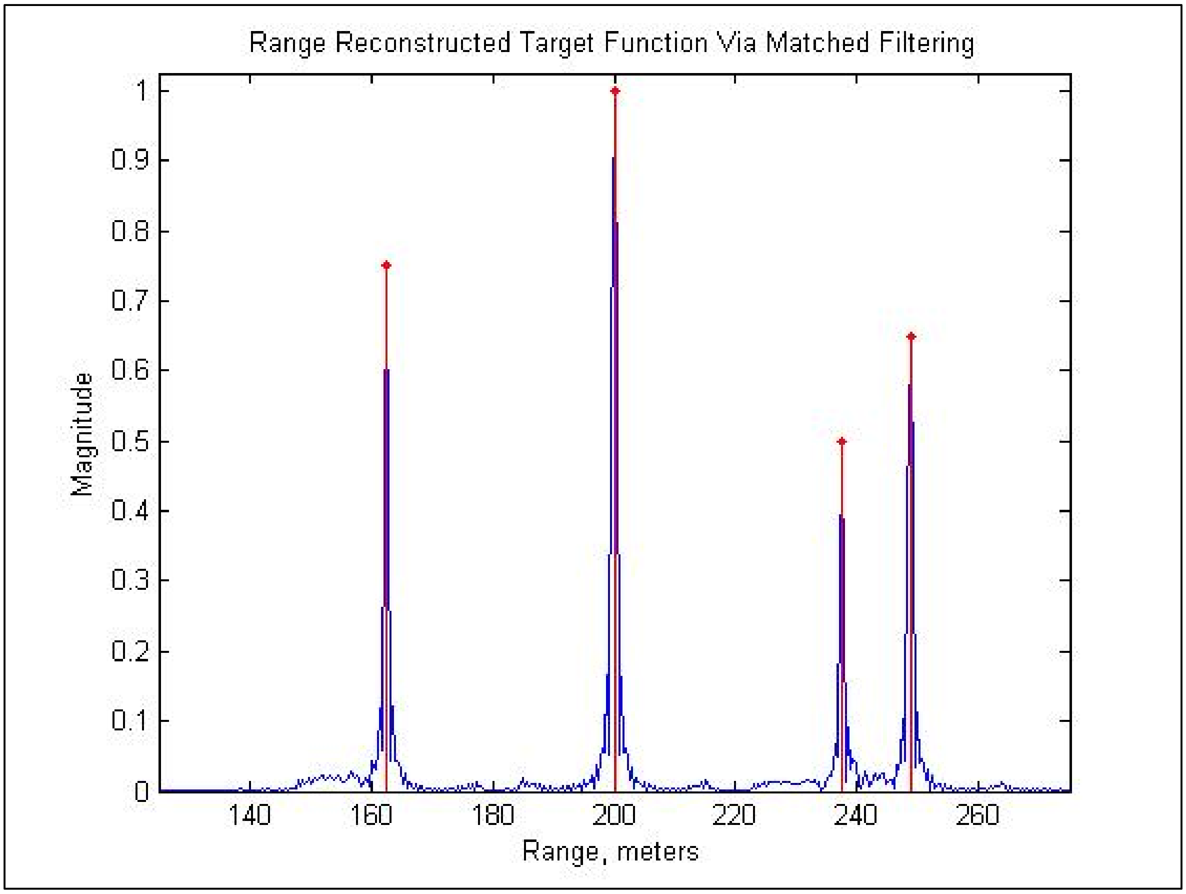

Figure 10. LFM Chirp Range Reconstruction via Matched Filtering.

4. White Gaussian Noise

The third signal type attempted for range imaging was white Gaussian noise. The noise was generated via the MATLAB function *randn.m*, which produces pseudo-randomly generated independent numbers with a normal distribution, zero mean, and unit variance. To facilitate testing, a global vector was populated with a ten thousand point sequence of noise. This sequence was passed to the *SigGen.m* function that defines the pulse duration and the same Boolean windowing function used in the case of the chirp. That windowing function is then applied to a portion of the noise sequence that is equal to the length of the input time vector, which is then upconverted as in (6). The output of this function is then subject to the same steps as the other signal forms; first baseband conversion and reference signal generation as can be seen in Figure 11 and Figure 12, then a matched filter is applied; the magnitude of the resulting signal is shown in Figure 13.

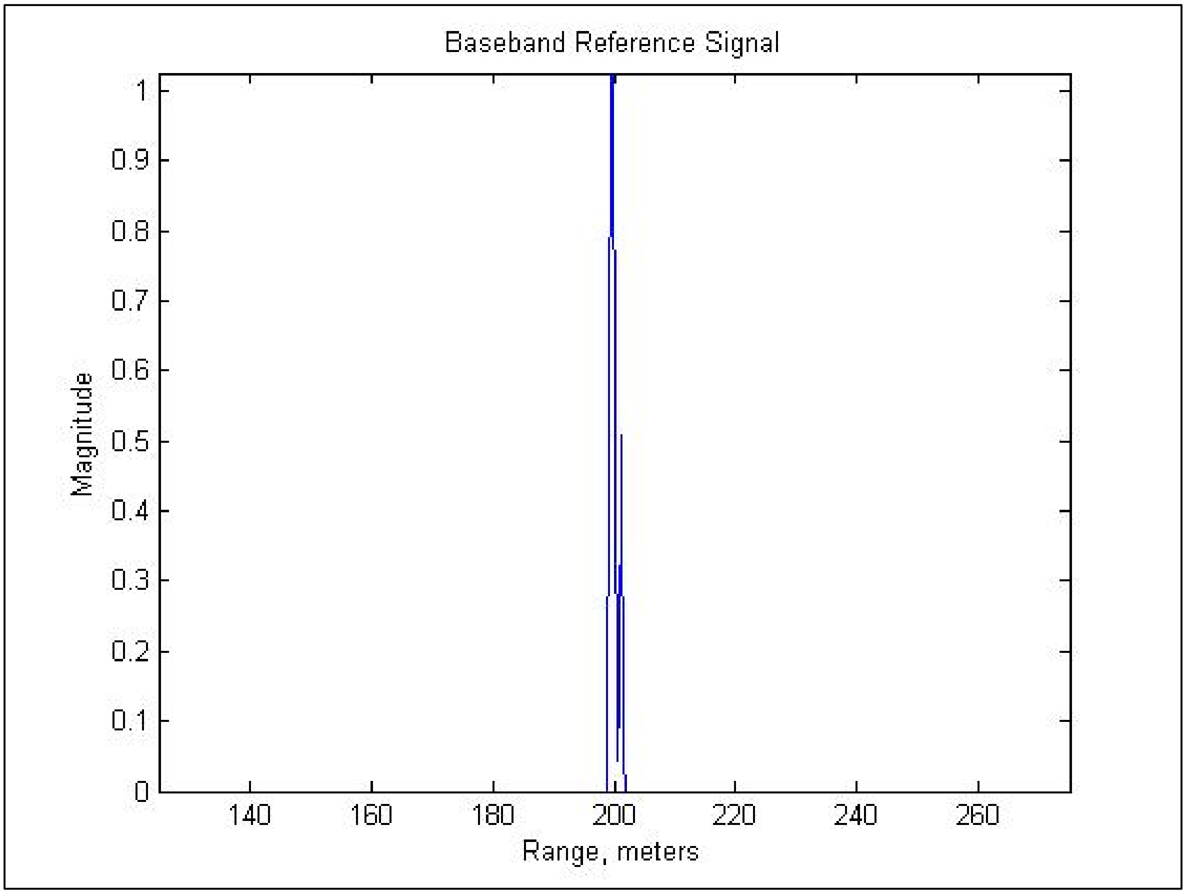

Figure 11. AWGN Baseband Reference Signal.

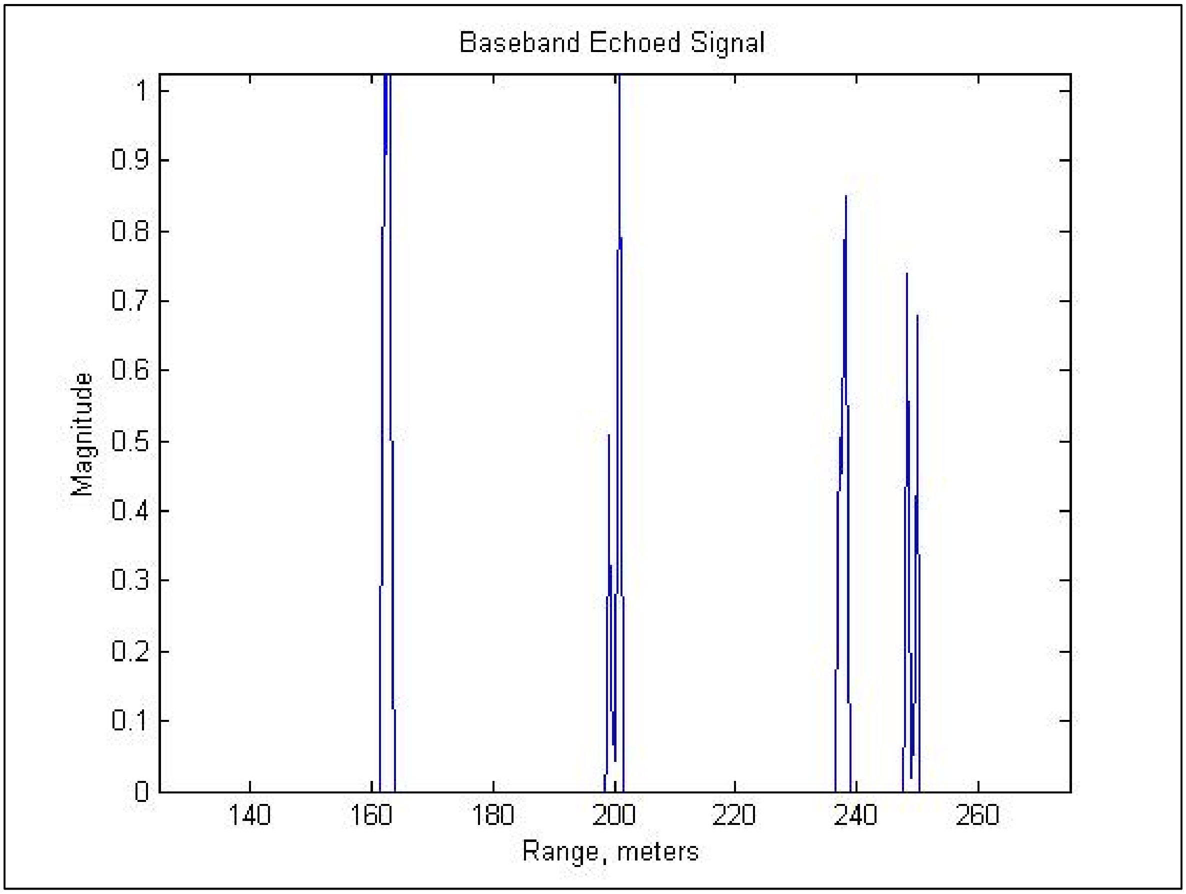

Figure 12. AWGN Baseband Echoed Signal.

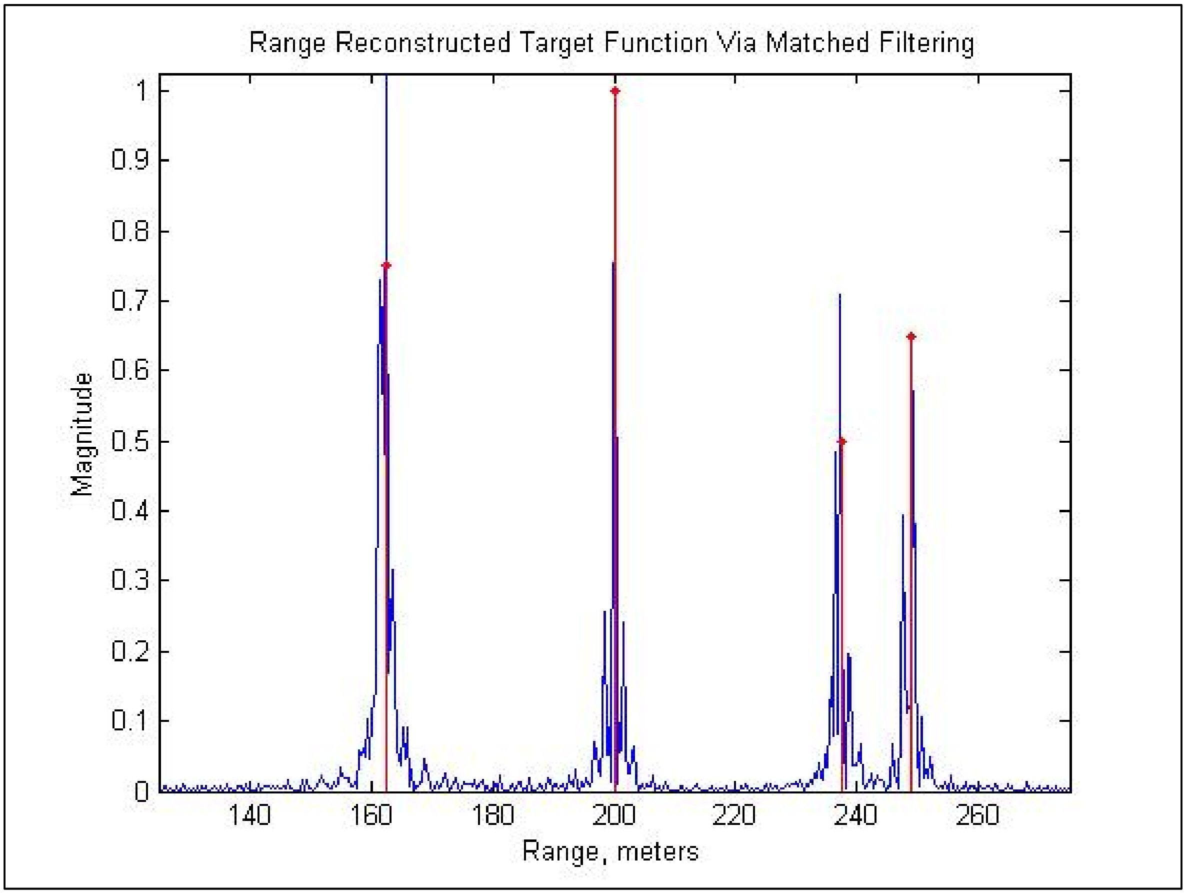

Figure 13. AWGN Range Reconstruction via Matched Filtering.

It can be seen in the figure above that when using white noise the ranges to the targets were accurately found. The drawback is that the amplitudes of each target successfully ranged do not correspond well to the target’s reflectivity. The transmitted signal’s power spectral density has a rectangular shape with bandwidth and parameters as follows:

- Pulse duration: $T_p = 10\text{ns}$
- Signal Energy: 38nJ
- Carrier Frequency: $f_c = 1\text{GHz}$
- Absolute Bandwidth: $\text{BW}=500\text{MHz}$

5. Range Imaging Summary

For the sake of continuity, the carrier frequency, pulse length, and target scene composition were kept constant for all three signal type simulations. The bandwidths and energies of the three signals used are similar, but not equal. All three signals correctly found the ranges of the targets. The LFM chirp had the best range resolution for the chosen simulation parameters, which can be seen by the minimal width of the main lobes in Figure 10. The noise pulse was able to accurately identify the range of the targets, but not the reflectivity.

All MATLAB code generated for this simulation is attached in Appendix A.

B. CROSS-RANGE IMAGING

In this section the synthetic aperture portion of SAR will be discussed. A synthetic aperture is a single transceiver (transmit/receive) that is physically moved over a distance. The transceiver could be mounted on numerous different platforms, with this simulation assuming that it is mounted on an aircraft. The distance traversed is $2L$ and referred to as the slow-time domain, given the designation of u in future discussion.[6] It is called the slow-time domain because the relative speed of the platform compared to the propagation speed of RF energy is very slow. It is because of this difference in speeds that, for the sake of the mathematics, it can be assumed that the platform stops and undergoes a full transmit/receive cycle then moves on again.

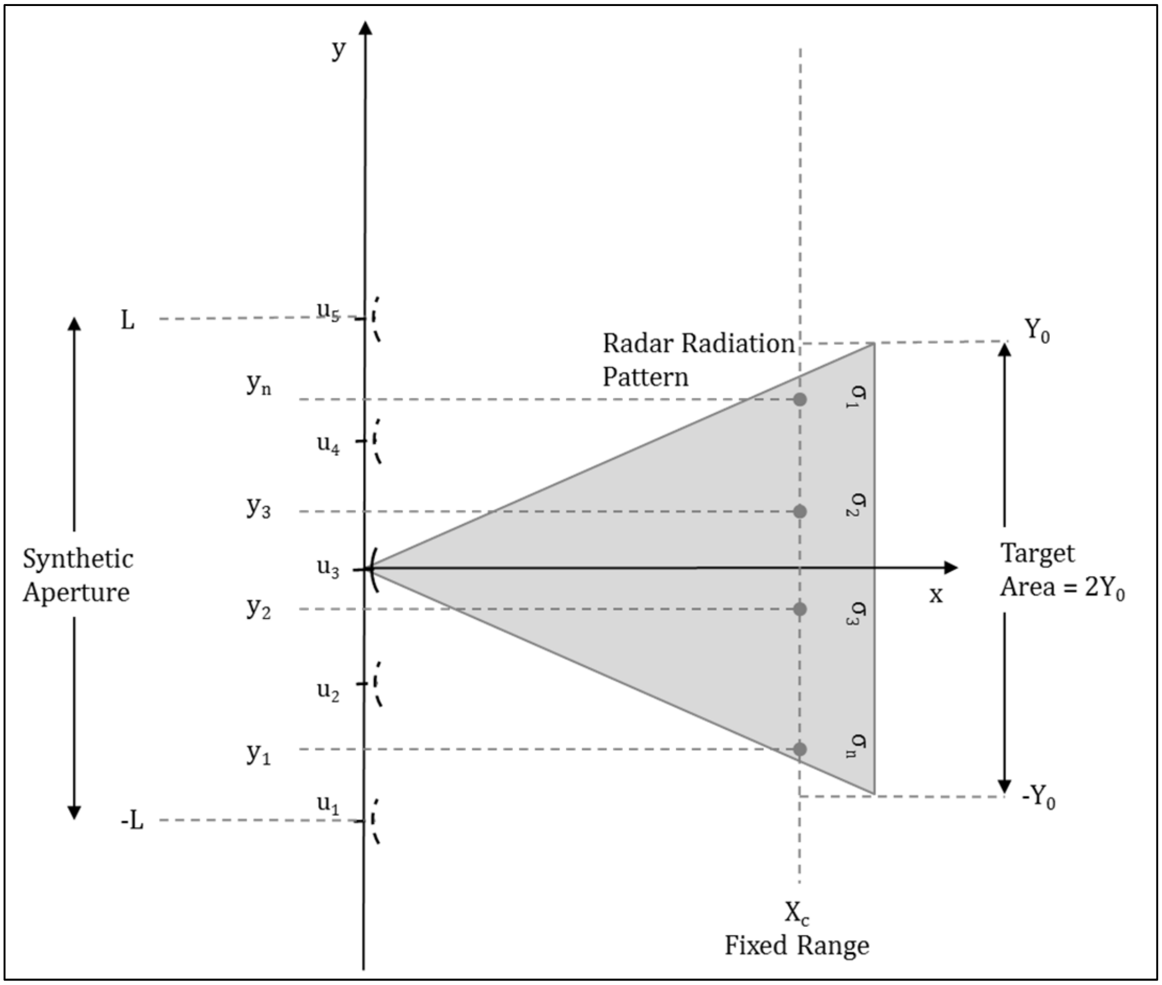

Figure 14. System Geometry for Cross-range Imaging after [6].

The target area for this simulation is assumed to be held to a broadside mode, where all the targets fall within the cross-range interval

$$y_n \in \left[-Y_0, Y_0\right], n = 1, 2, \dots \tag{13}$$

and within the synthetic aperture limited by

$$\left|y_n\right| < L. \tag{14}$$

The final assumption made is that all targets are at the same range, and parallel to the platform’s path of motion, i.e., $x_n = X_c \left(n = 1, 2, \ldots\right)$, where X_c is a known constant leaving the target locations of the form $\left(X_c, y_n\right)$.

The above assumptions are made to reduce the complexity of the equations and simplify the code used.

1. General Case

It is useful to define an ideal target function in the same manner as was done for range imaging, i.e. with lumped targets a represented using the Dirac delta function

$$f_0\left(y\right) = \sum_n \sigma_n \delta\left(y - y_n\right), \qquad \textbf{(15)}$$

where σ_n is the nth target reflectivity. The distance between the radar at $\left(0, u\right)$ and the target located at $\left(x_n, y_n\right)$ is

$$\sqrt{x_n^2 + \left(y_n - u\right)^2}. \qquad \textbf{(16)}$$

By examining the case where the transmitted signal is a single tone pulse given by

$$p\left(t\right) = e^{j\omega t} I_{\left[0, T_p\right]}\left(t\right), \qquad \textbf{(17)}$$

and as u varies the recorded echoed signal can be expressed as

$$s\left(t, u\right) = \sum_n \sigma_n p\left[t - \frac{2\sqrt{x_n^2 + \left(y_n - u\right)^2}}{c}\right]. \qquad \textbf{(18)}$$

Combining equations (17) and (18) yields echoed signal

$$s\left(t,u\right)=\exp\left(j\omega t\right)\sum_{n}\sigma_{n}\exp\left[-j2k\sqrt{x_{n}^{2}+\left(y_{n}-u\right)^{2}}\right]I_{\left[0,T_{p}\right]}\left(t\right), \quad \textbf{(19)}$$

with $k=\omega/c$ being the wavenumber. Applying a baseband conversion to (19) the final reflected signal

$$\begin{aligned} s\left(\omega,u\right)&=s\left(t,u\right)\exp\left(-j\omega t\right)\\ &=\sum_{n}\sigma_{n}\exp\left[-j2k\sqrt{x_{n}^{2}+\left(y_{n}-u\right)^{2}}\right]I_{\left[0,T_{p}\right]}\left(t\right) \end{aligned} \quad \textbf{(20)}$$

is produced.

2. Cross-range Reconstruction via Matched Filtering

Cross-range reconstruction is very similar to range imaging reconstruction. The received baseband signal $s\left(\omega,u\right)$ undergoes a spatial Fourier transform (i.e. with respect to u), like the time domain transform in the range imaging case. The individual reflected signals, $s_{n}\left(\omega,u\right)$, undergo a spatial Fourier transform and are summed together to form the total reflected signal

$$\begin{aligned} S\left(\omega,k_{u}\right)&=S\left(ck,k_{u}\right)\\ &=\sum_{n}\sigma_{n}S_{n}\left(\omega,k_{u}\right)\\ &=\sum_{n}\sigma_{n}\exp\left(-j\sqrt{4k^{2}-k_{u}^{2}}x_{n}-jk_{u}y_{n}\right) \end{aligned} \quad \textbf{(21)}$$

for $k_{u}\in\left[-2k,2k\right]$ and zero otherwise with $k=\frac{\omega}{c}$.[6] The reference signal is generated from a unit target reflector at the center of the target area and given by

$$s_{0}\left(\omega,u\right)=\exp\left[-j2k\sqrt{X_{c}^{2}+u^{2}}\right]. \quad \textbf{(22)}$$

The reconstructed target function is produced by the matched filtering

$$F\left(k_y\right)=S\left(\omega,k_u\right)S_0^*\left(\omega,k_u\right) \\ \sum_n \sigma_n \exp\left(-jk_y y_n\right) \tag{23}$$

where S_0^* is the complex conjugate of s_0, the cross-range spatial frequency mapping is $k_y=k_u$, and $k_y\in\left[-2k,2k\right]$ from [6]. An inverse Fourier transform is then applied to return the signal to the spatial synthetic aperture domain.

3. Cross-range Imaging Simulation

The simulation was coded in MATLAB and influenced heavily by the *crange.m* script that was included in [6]. The signal generation and all pertinent signal properties were removed from the main simulation script and compose the signal generation function. In most imaging applications, the highest resolution possible is desired. The transmitted signal is of a sinusoidal pulse and has a Fourier transform of the $\frac{\sin\left(x\right)}{x}$ shape. Depicted in Figure 15 is the baseline for this simulation in terms of resolution and was generated using the following parameters for the transmitted radar pulse and the system geometry:

- Carrier Frequency: $f_c = 100\text{MHz}$
- Pulse Duration: $T_p = 535\text{ns}$
- Null-to-Null Bandwidth: $BW = 935\text{KHz}$
- Signal Energy: 535nJ
- Synthetic Aperture: $2L = 800\text{m}$
- Range to Center of Target Area: $X_c = 500\text{m}$

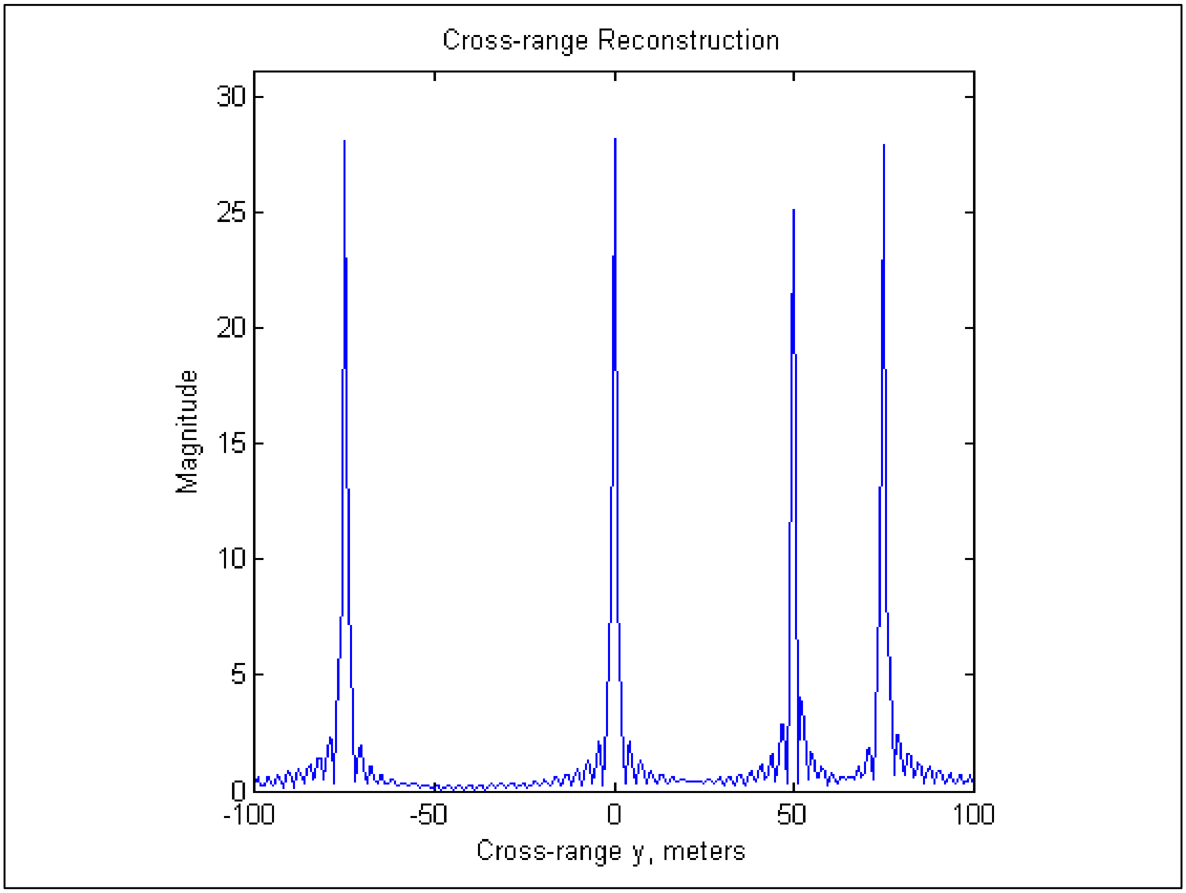

Figure 15. Cross-range Reconstruction via Matched Filtering – Normal Resolution.

Per [6], there are four factors that determine the bandwidth in this simulation and thus affect the resolution of the reconstruction. The first is the size of the synthetic aperture *2L*. As the aperture increases so does the bandwidth and correspondingly the resolution. The second parameter that affects the resolution is the radar frequency. The third is the target range given in (16), and the fourth is not applicable due to the assumption that the target is always a broadside one. The reason that these three factors affect the cross-range resolution is that they artificially increase the effective bandwidth of the transmitted signal through the slow-time Doppler filter. By utilizing a larger effective bandwidth the signal space is capable of a greater number of samples.

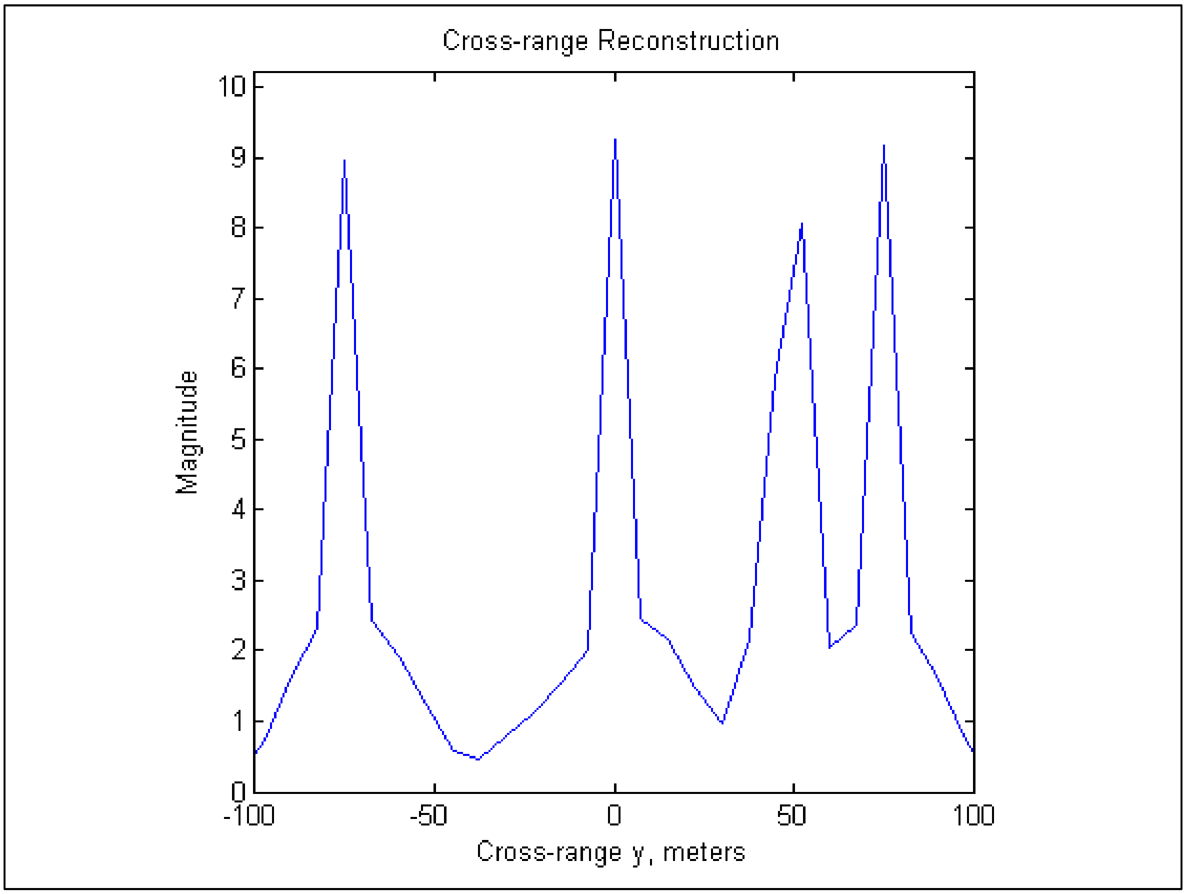

Figure 16. Cross-range Reconstruction via Matched Filtering – Low Resolution.

Illustrated in Figure 16 is the reduction in resolution from moving the target area further away from the synthetic aperture, resulting in fewer samples than the baseline case and subsequently a lower resolution. The parameters used to generate it are as follows:

- Carrier Frequency: $f_c = 100\text{MHz}$
- Pulse Duration: $T_p = 550\text{ns}$
- Null-to-Null Bandwidth: $BW = 909\text{KHz}$
- Signal Energy: 550nJ
- Synthetic Aperture: $2L = 800\text{m}$
- Range to Center of Target Area: $X_c = 5000\text{m}$

In Figure 17, a greater resolution than the baseline case is illustrated through increasing the carrier frequency, which increases the number of samples taken. The parameters used to generate it are as follows:

- Carrier Frequency: $f_c = 500\text{MHz}$
- Pulse Duration: $T_p = 2.67\text{ms}$
- Null-to-Null Bandwidth: $BW = 187\text{KHz}$
- Signal Energy: 2.67mJ
- Synthetic Aperture: $2L = 800\text{m}$
- Range to Center of Target Area: $X_c = 500\text{m}$

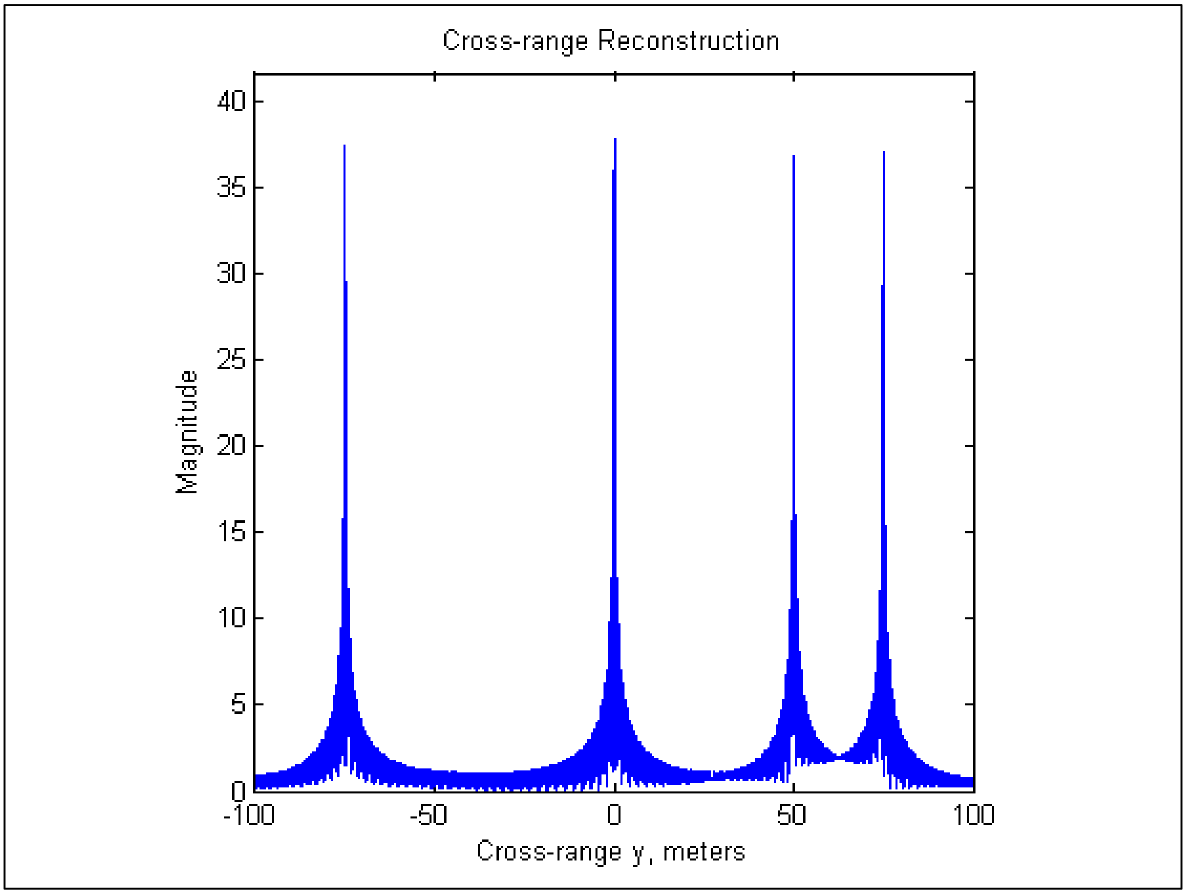

Figure 17. Cross-range Reconstruction via Matched Filtering – High Resolution.

4. Cross-range Imaging Summary

This section has demonstrated the importance that the physical parameters possess when dealing with the cross-range imaging component of SAR. The three parameters that were examined in this section were carrier frequency, distance to target area center, size of the synthetic aperture. By increasing the carrier frequency or size of the synthetic aperture, which is how far the craft bearing the transceiver moves while undergoing transmit/receive cycles, the bandwidth of the slow-time Doppler filter increases as well, resulting in better resolution.

Conversely, by reducing the distance to the center of the target area a similar effect is achieved, again increasing the effective bandwidth of the slow-time Doppler filter utilized thus increasing resolution. Increasing the carrier frequency also increases the signal energy required to gain the higher resolution, just like the other methods of obtaining a higher resolution have a penalty associated with them. This simulation is also conducted in a sterile "environment" meaning there is no attenuation or ambient noise accounted for to interfere, making these results the best case scenario.

All MATLAB code generated for this simulation is attached in Appendix B.

C. SYNTHETIC APERTURE RADAR PROCESSING

1. Stripmap Synthetic Aperture Radar

Now that the Range and Cross-range imaging system models have been detailed, the next step is to combine the two into the generic synthetic aperture radar model. The model that is used is called Stripmap SAR. It is named this due to the linear motion of the transmitting platform. The target is composed of a set of point reflectors that are stationary and are located in the spatial domain at coordinates $(x_n, y_n), n = 1, 2, \ldots$. The radar aperture, or flight path, resides along $(0, u)$ and illuminates the target area with the signal $p(t)$.[6]

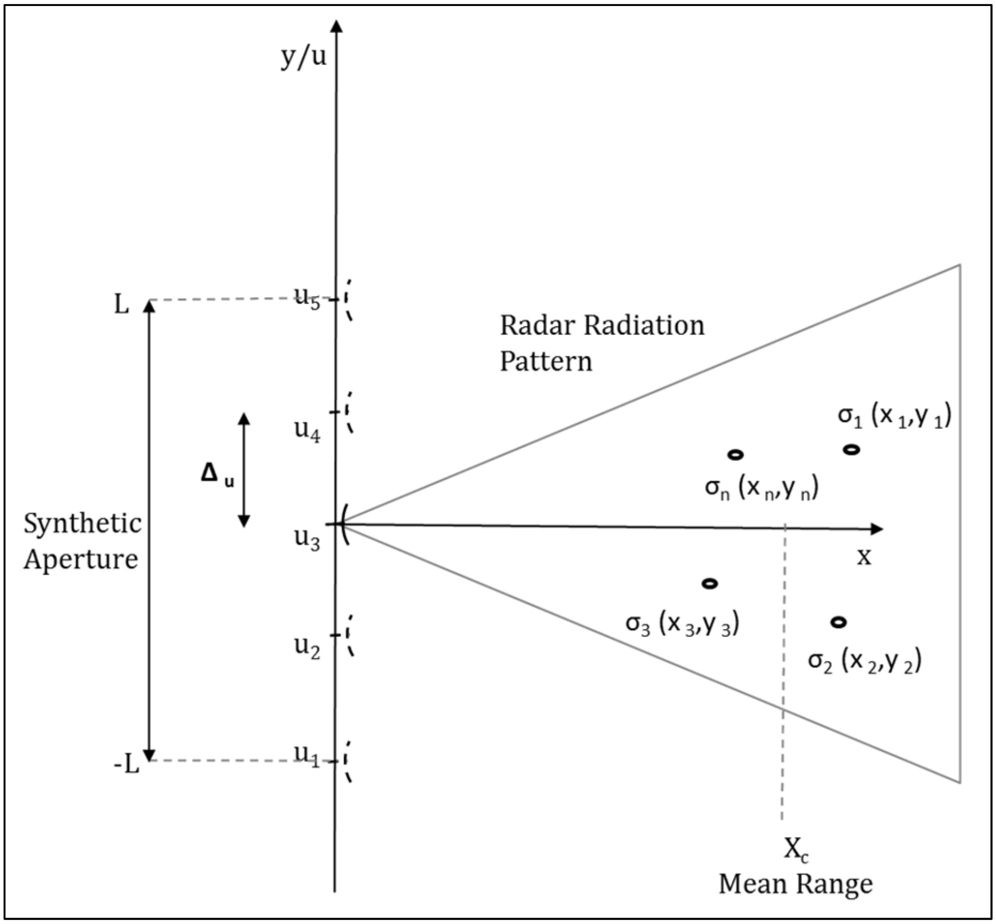

Figure 18. Stripmap SAR System Geometry. After [6].

The signal bandwidth is determined by which signal type is selected, with the linear frequency modulated chirp signal being the most common for current SAR applications. This implementation of SAR transforms a three variable problem into a two variable one through the use of the slant range, which is the distance from the aircraft to the ground target calculated using the Pythagorean Theorem. The measured signal return is

$$s(t,u)=\sum_n \sigma_n p\left[t-\frac{2\sqrt{x_n^2+(y_n-u)^2}}{c}\right], \tag{24}$$

where

$$\frac{2\sqrt{x_n^2+(y_n-u)^2}}{c} \tag{25}$$

is the round trip time for the signal to propagate to the n^{th} target and back From [6].

The reflected signal then undergoes reconstruction via fast-time matched filtering given by

$$\begin{aligned}\Im_t\Im_u\left[s(t,u)\right] &= F\left[k_x(\omega,k_u),k_y(\omega,k_u)\right]\\ &= P^*(\omega)S(\omega,k_u)\\ &=|P(\omega)|^2\sum_n \sigma_n e^{\left(-jk_x x_n-jk_y y_n\right)}\end{aligned} \tag{26}$$

where

$$\begin{aligned}k_x(\omega,k_u)&=\sqrt{4k^2-k_u^2}\\ k_y(\omega,k_u)&=k_u\\ \Im_t &\rightarrow \text{Time based Fourier transform}\\ \Im_u &\rightarrow \text{Spatial Fourier transform}\\ F[\;\;] &\rightarrow \text{Reconstructed target function}\end{aligned} \tag{27}$$

with k_u being the spatial frequency or wavenumber for the synthetic aperture position u.

2. Stripmap SAR Simulation Rectangular Sinusoidal Pulse

Once the validity of the simulation was confirmed, the signal generation portion was relegated to its own function. This was done to better understand the effects of different signal types on the reconstructed SAR image. The first signal type studied using this simulation was a sinusoidal rectangular pulse. The magnitude of the reflected signal $|s(t,u)|$, and correspondingly the magnitude of that signal after matched filtering, $|f(t,u)|$ are displayed in Figure 19 and Figure 20.

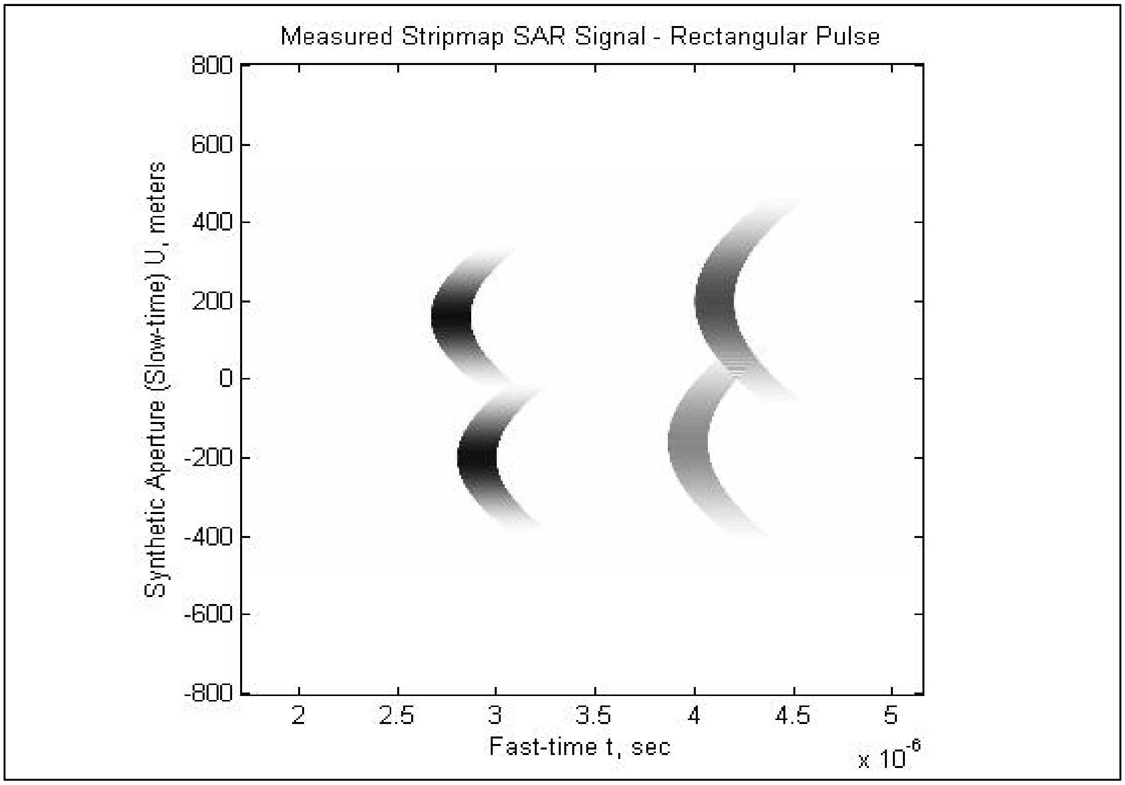

Figure 19. Stripmap SAR Signal using a Rectangular Sinusoidal Pulse.

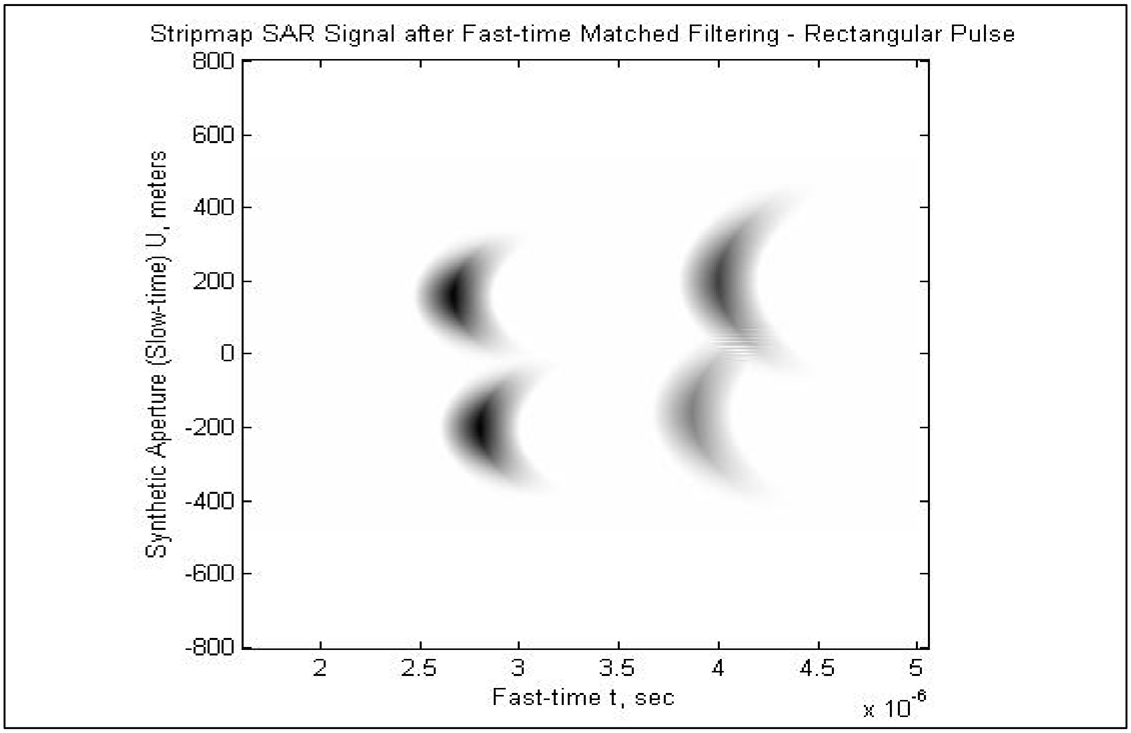

Figure 20. Stripmap SAR Signal after Fast-time Matched Filtering using a Rectangular Sinusoidal Pulse.

The transmitted signal has a Fourier Transform in the shape of a sinc function with relevant parameters:

- Sinusoidal of the form $e^{j2\pi f_c t} I_{\left[0,T_p\right]}\left(t\right)$
- Pulse duration: $T_p = 200\text{ns}$
- Carrier Frequency: $f_c = 1.5\text{GHz}$
- Signal Energy: 11.6mJ
- Null-to-Null Bandwidth: $2f_0 = 100\text{MHz}$
- Distance to Center of Target Area: $X_c = 500\text{m}$
- The width of the target space in the range domain: $2x_0 = 400\text{m}$
- The length of the target space in the cross-range domain: $2y_0 = 800\text{m}$

3. Stripmap SAR Simulation LFM Chirp

The next signal type examined was a linearly frequency modulated chirp. The LFM chirp signal is more desirable in synthetic aperture radar applications because of the larger usable bandwidth that is available for the same pulse duration. For this part of the SAR simulation an "up" chirp was used, meaning that the frequency was increasing following the form of (11). The larger bandwidth relates to a better resolution which can be seen in the magnitude of the reflected signal $\left|s\left(t,u\right)\right|$, and then the magnitude of the reconstructed signal $\left|f\left(t,u\right)\right|$ after matched filtering illustrated in Figure 21 and Figure 22.

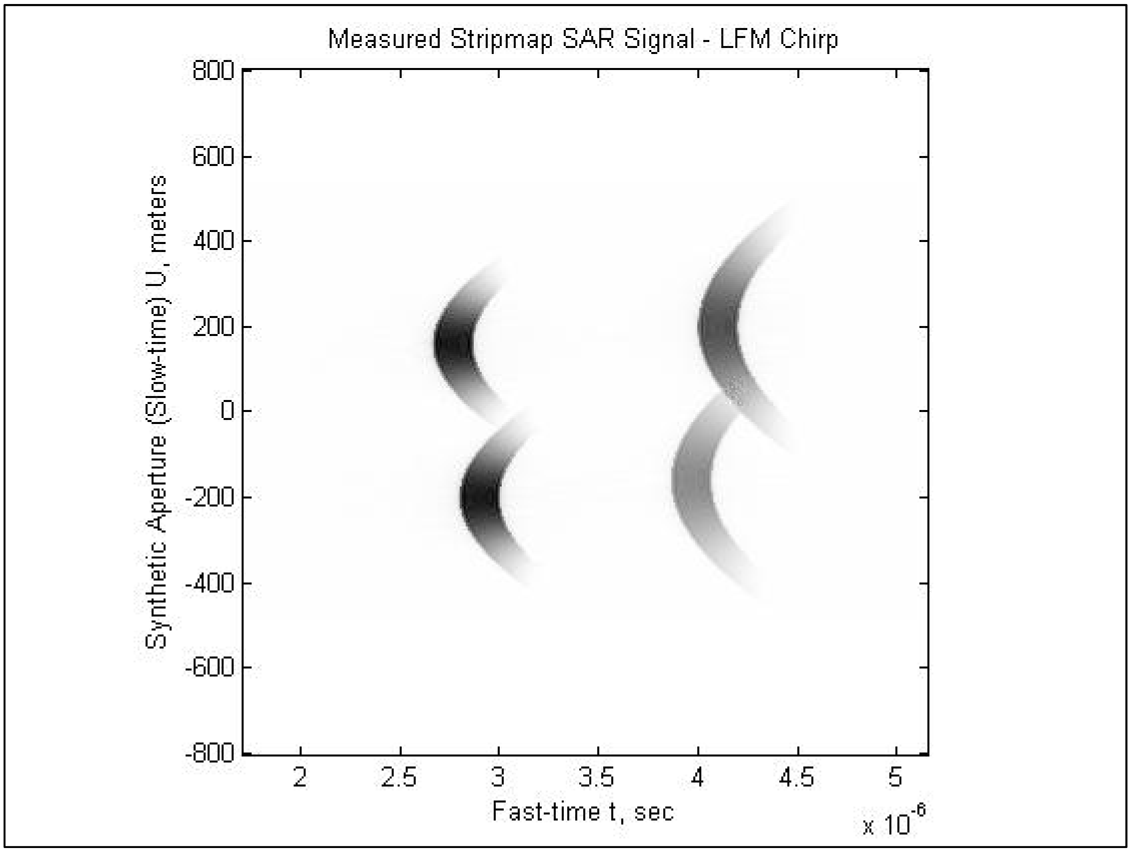

Figure 21. Stripmap SAR Signal using a Linearly Frequency Modulated Chirp.

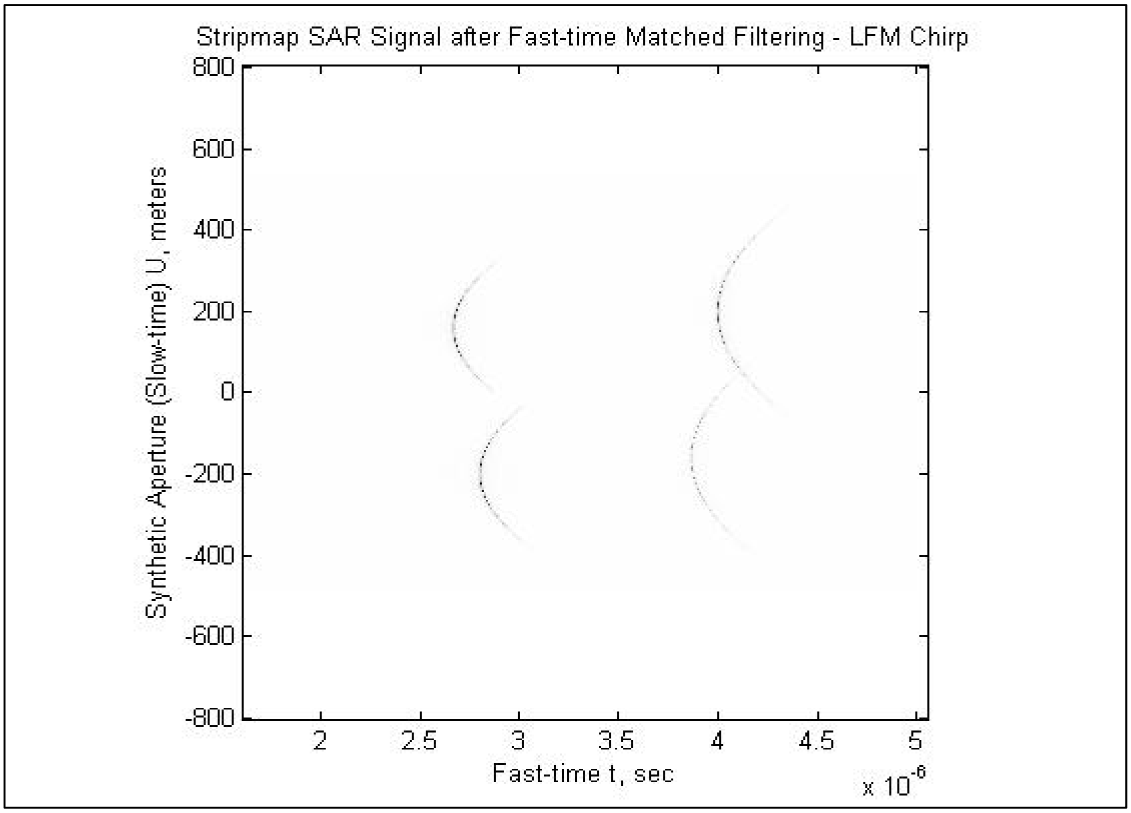

Figure 22. Stripmap SAR Signal after Fast-time Matched Filtering using a Linearly Frequency Modulated Chirp.

The difference in resolution between the sinusoidal pulse and the LFM chirp signal, when used in the SAR simulation, is significant and is easily discernible when comparing Figure 20 and Figure 22. The LFM chirp bandwidth was calculated from the range of instantaneous frequencies. All relevant parameters to recreate this section of the simulation are:

- LFM Chirp given by (11)
- Chirp rate (α): $2\pi\times10^{16}\,\text{rad/s}^2$
- Pulse duration: $T_p = 200\text{ns}$
- Carrier Frequency: $f_c = 1.5\text{GHz}$
- Signal Energy: 11.6mJ
- Null-to-Null Bandwidth: $\frac{\alpha T_p}{2\pi} = 2\text{GHz}$
- Distance to Center of Target Area: $X_c = 500\text{m}$
- The width of the target space in the range domain: $2X_0 = 400\text{m}$
- The length of the target space in the cross-range domain: $2Y_0 = 800\text{m}$

4. Stripmap SAR Simulation Band-Limited Gaussian White Noise

The third signal type implemented in the SAR imaging simulation was white Gaussian noise. The noise was again generated via the MATLAB function *randn.m*, which produces the pseudo-randomly generated independent numbers that have a normal distribution, zero mean, and unit variance. To facilitate the testing, a global vector was populated with a one hundred thousand point sequence of noise. This sequence was passed to the signal generation function that defines the pulse duration and the same Boolean windowing function used in the case of the chirp. That windowing function is then applied to a portion of the noise sequence that is equal in length to the input time vector, which is then upconverted by (6). The results of the Gaussian noise as an input to the SAR simulation can be seen in Figure 23 and Figure 24.

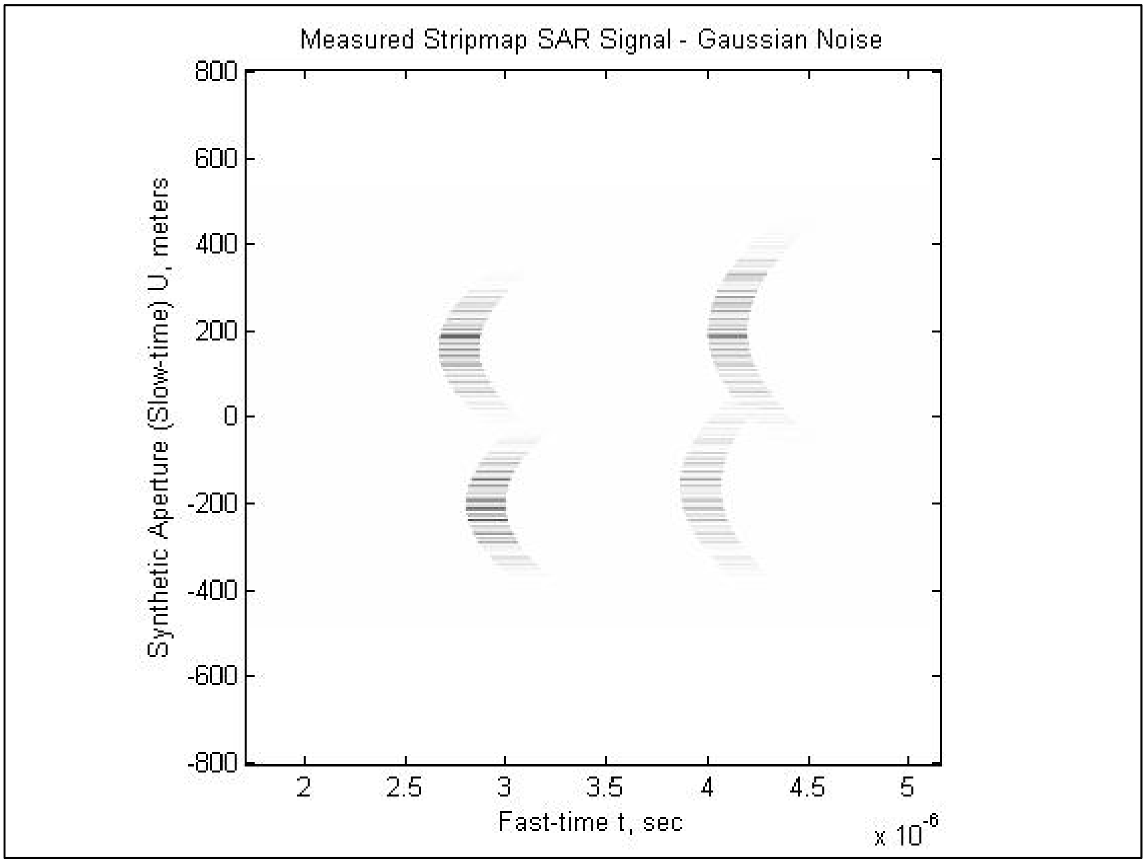

Figure 23. Stripmap SAR Signal using Gaussian Noise.

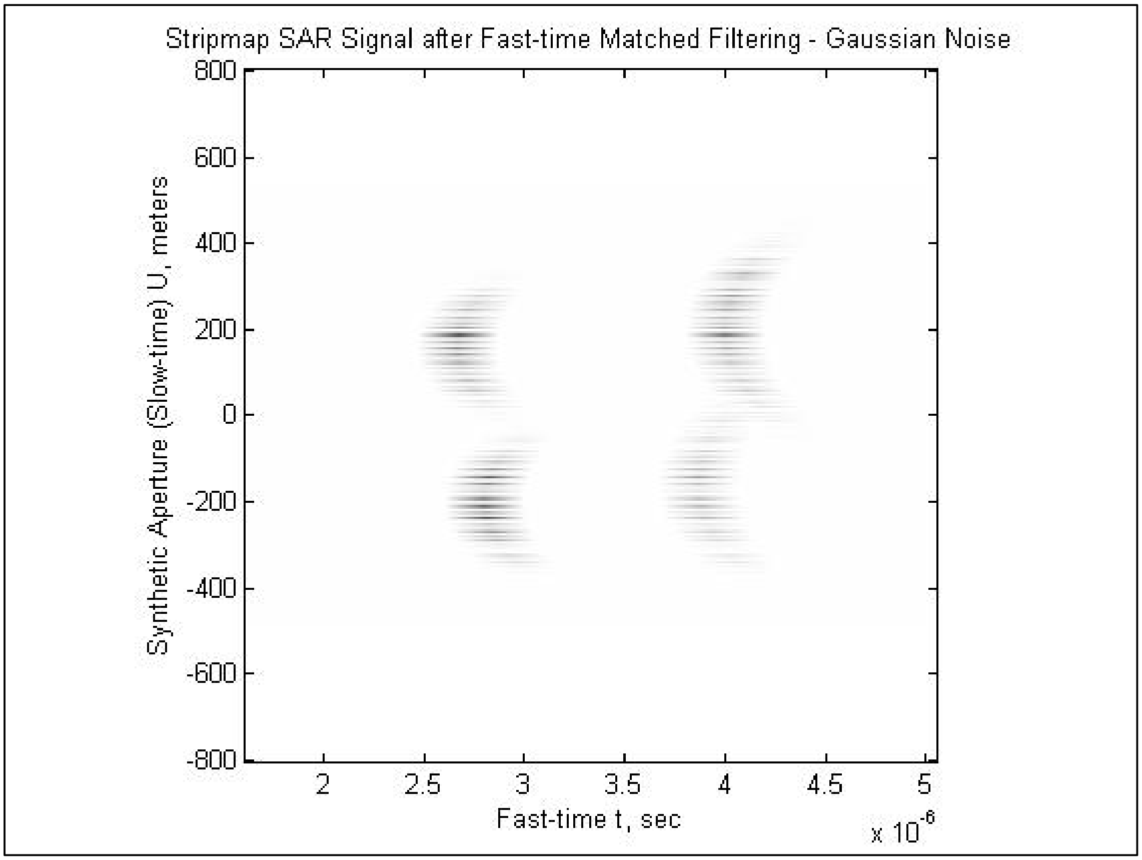

Figure 24. Stripmap SAR Signal after Fast-time Matched Filtering using Gaussian Noise.

The effect the noise, as an input signal, has on the resulting SAR image is illustrated in Figure 24. The coherence of the generated image indicates a promising area for future research. The SAR simulation parameters for the Gaussian noise signal type are as follows:

- Pulse duration: $T_p = 200\text{ns}$
- Carrier Frequency: $f_c = 1.5\text{GHz}$
- Signal Energy: 34.7mJ
- Absolute Bandwidth: $2f_0 = 400\text{MHz}$
- Distance to Center of Target Area: $X_c = 500\text{m}$
- The width of the target space in the range domain: $2x_0 = 400\text{m}$
- The length of the target space in the cross-range domain: $2y_0 = 800\text{m}$

5. Stripmap SAR Simulation Summary

In the SAR model, the pulse duration, carrier frequency, distance to center of target area, and length and width of the target space were held constant. The signal energy and null-to-null bandwidth varied according to the signal type selected. The LFM chirp signal created the image with the highest resolution of the three test cases. It can be seen in Figure 22 the precision achieved by how fine the arc is. The sinusoidal pulse signal input resulted in the second best resolution as can be seen in Figure 20, with the targets being represented by a thicker, solid arc. The signal energies of the LFM chirp and the sinusoidal pulse were calculated to be approximately equal, within 80nJ of one another, though the bandwidths of the signals were very different. This is due to the additive nature of the LFM signal properties and the slow-time Doppler filter that was discussed in the cross-range imaging section. It is this disparity in bandwidth that causes the large difference in image resolutions.

The final signal input type that the SAR model was subjected to is the band-limited Gaussian white noise. As in the range imaging simulation, the noise input signal correctly returned the positions of all the targets with perfect accuracy. The resolution of the generated image is almost at the same level as the sinusoidal pulse signal input, the difference is in the overall fidelity of each target returns. The signal energy for the noise tests were the highest of the three, while the bandwidth was in the middle. This leads to the conclusion that, although it appears to be a valid input signal type, it is far less efficient than the other signal types studied. The results generated from the noise testing with the SAR model provide an interesting line of future research such as, attempting to overcome the lack of precision in the amplitude of the returned signal, increasing the efficiency, and studying the effects colored noise would have on the simulation results.

All MATLAB code generated for this simulation is attached in Appendix C.

THIS PAGE INTENTIONALLY LEFT BLANK

III. CONCLUSION AND FUTURE RESEARCH

A. CONCLUSION

1. Synthetic Aperture Radar

Synthetic aperture radar is a crucial part of the imagery intelligence world, not only for military applications but also for numerous civilian ones as well. Although the technology has been operational for a long time, the potential uses and creative applications of SAR and the different signal reconstruction algorithms have yet to be explored in their entirety leaving room for growth and development in this field. The research presented here has demonstrated a number of key elements of the basis of SAR including a disassembled view of the components of producing SAR imagery, the effects differing signal types have on image resolution and accuracy, and the intriguing prospect of using noise itself as a signal. By breaking SAR down into range imaging and cross-range imaging building blocks it is far easier to understand the physics and signal processing of SAR.

The range imaging section demonstrated that the LFM chirp signal produced the highest resolution for the constant parameters of the test case. This is expected since the nature of the LFM chirp signal allows for a greater resolution for the same length of pulse as used in the rectangular pulsed signal. The largely unexpected results from the range imaging construct was the accuracy returned by the band-limited Gaussian white noise. The lack of fidelity observed in the amplitudes of the returned signal is disappointing and could possibly be compensated for in future research.

The cross-range imaging section established the importance that the physical components of SAR, distance of target scene from the craft, size of synthetic aperture, and carrier signals introduce into the overall SAR calculations while minimalizing the effects of differing signal types. The reason that the signal type itself has a minimal impact on the cross-range imaging results is, in short, due to the slow-time Doppler filter used in the simulation. The following parameters directly influence the resolution in the image, the platform's orientation or aspect angle to the target area, the distance to the target area, the carrier frequency, and the size of the aperture being synthesized by the crafts motion. Each of the above listed parameters affect the resolution of the reconstructed image significantly more than the signal type input as discussed in the cross-range imaging section.

The final portion of the thesis synthesizes the range and cross-range concepts to form the desired result of a working Synthetic Aperture Radar simulation. Taking the lessons learned in the SAR component pieces of the research it was shown, with an expected level of consistency, that resolution is directly a function of the actual or synthesized signal or filter bandwidth provided by the signal type, carrier frequency, target composition, length of aperture, and target aspect ratio. The LFM chirp input signal again provided the highest resolution with a standardized target composition and synthetic aperture length. The sinusoidal pulse and the band-limited Gaussian white noise signal inputs returned almost identical results with the resolutions and fidelity of the target area being represented in the generated images with only small variances between them. The introduction of noise as the SAR input signal itself shows an interesting potential for what other signal types could achieve when paired with the matched filtering signal reconstruction model presented.

B. FUTURE RESEARCH

1. Continuing Research

There are a number of areas for continued research on this subject. The first would be research into the multiple digital reconstruction methods and how they perform when using different signal types as input, with particular attention being paid to the bandwidth used in each and compared against the resolution achieved. The second would be further research into using noise as an input signal itself and the effects of using colored noise versus white noise, along with constructing an algorithm to compensate for the lack of fidelity in regards to the reflectivities of the targets illuminated when using noise itself as a signal. Another area of continued research would be to modify the simulation to emit a rapid burst of identical or slightly varying pulses and comparing the resultant resolution against using only one pulse per generated image. Finally, these simulations should be repeated to compare various signal types while maintaining equal signal energies, bandwidths, pulse durations and carrier frequencies and also repeated with a single signal type while varying one of the above parameters at a time to qualitatively determine the relationships between these parameters and the quality of the image.

APPENDIX A

The MATLAB code contained in Appendix A is used to generate the figures used to illustrate the results presented in the Range Imaging section.

Range Imaging Simulation

```
clear all;

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% ASSUMPTIONS: Tx > Tp;                                 %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

% Define constants and signal parameters
global NOISE;
NOISE=randn(1,10000);
c=3e8; pi2=2*pi; b0=100e6; w0=pi2*b0;
sigType = input(‘2 = AWGN, 1 = LFM Chirp, 0 = Rect Pulse: ‘);        %
1 for LFM Chirp, 0 for Rect Pulse
% sigType = 1 ;
[y,wc,wcm,Tp,alpha]=R_SigGen_v2(0,w0,sigType);

% Initialize target space, sample spacing and characteristics
[ x,xc,x0,t,w,n ] = R_initialize( w0,alpha,b0,c,Tp,pi2,wc );

% Target definitions
ntarget=7;      % used only when ident=1
ident = 1;      % set ident to 1 for random target gen
                % ident=0 is static 4 target generation
if ident==0
    ntarget=4;  % test case for static vs. dynamic target #s
end
[xn,fn]=R_targets(ident,x0,ntarget);

% initialize reflected signal
s=zeros(1,n);

% reflected signal calculation
for i=1:ntarget
    td=t-2*(xc-xn(i))/c;
    s = s + fn(i)*R_SigGen_v2(td,w0,sigType);
end

% Reference Signal calculation
td0=t-2*(xc+0)/c;
s0=R_SigGen_v2(td0,w0,sigType);

% Test to determine which case and bb conversion as appropriate
if sigType == 1
```

```
    % Baseband conversion
    sb=s.*exp(-1i*wcm*t);
    sb0=s0.*exp(-1i*wcm*t);

    figure(1)
    plot(t,fliplr(real(sb)))
    xlabel(‘Time, seconds’)
    ylabel(‘Real Part’)
    title(‘Baseband Echoed Signal’)

    figure(2)
    plot(t,real(sb0))
    xlabel(‘Time, seconds’)
    ylabel(‘Real Part’)
    title(‘Baseband Reference Signal’)
else
    % Baseband conversion
    sb=s.*exp(-1i*wc*t);
    sb0=s0.*exp(-1i*wc*t);

    figure(1)
    plot(x,fliplr(abs(sb)))
    xlabel(‘Range, meters’)
    ylabel(‘Magnitude’)
    axis([xc-x0 xc+x0 0 1.025])
    title(‘Baseband Echoed Signal’)

    figure(2)
    plot(x,abs(sb0))
    xlabel(‘Range, meters’)
    ylabel(‘Magnitude’)
    axis([xc-x0 xc+x0 0 1.025])
    title(‘Baseband Reference Signal’)
end

% Matched Filtering
fsb=fty(sb);
fsb0=fty(sb0);
fsmb=fsb.*conj(fsb0);

% Power equalization and windowing
E=R_PwrEq( fsb0 );

figure(3)
plot((w-wc)/pi2,abs(fsmb))
xlabel(‘Frequency, Hertz’)
ylabel(‘Magnitude’)
title(‘Baseband Match-Filtered Signal Spectrum’)

smb=ifty(fsmb);

figure(4)
plot(x,fliplr((n/E)*abs(smb)))
xlabel(‘Range, meters’)
```

```
ylabel(‘Magnitude’)
axis([xc-x0 xc+x0 0 1.025])
title(‘Range Reconstructed Target Function Via Matched Filtering’)
pause(1)
hold on
stem(xn+xc,(fn),’r.-’)
hold off
```

Signal Generation

```
function  [y, wc, wcm, Tp, alpha] = R_SigGen_v2(td,w0,sigType)
global NOISE;

fc = 1e9;
wc = 2*pi*fc;
alpha = 0;
wcm = wc;
n1=floor(length(NOISE)/length(td));
noise=[];
if sigType == 2
    for i=1:length(td)
        noise = [noise NOISE(((i-1)*n1)+1)];
    end
    Tp = .01e-6;
    ybb = 1*(td>=0 & td<=Tp);
    y = ybb.*noise;
    y = y.*exp(1i*wc*td);
elseif sigType == 1
    Tp = .01e-6;
    alpha = w0/Tp;
    wcm = wc-alpha*Tp;
    ybb = 1*(td>=0 & td<=Tp);
    y = ybb.*exp(1i*wcm*td+1i*alpha*(td.^2));
else
    Tp = .01e-6;
    ybb = 1*(td>=0 & td<=Tp);
    y = ybb.*exp(1i*wc*td);
end
```

Initialization

```
function [ x,xc,x0,t,w,n ] = R_initialize( w0,alpha,b0,c,Tp,pi2,wc )
%INITIALIZE Range imaging variables
% Target space definition
xc=200;                % Range to center of target area
x0=75;                % Target area [xc-x0,xc+x0]
Tx = (4*x0)/c          % Range swath echo time period

% Sample spacing
dt=pi/(2*w0);           % Time Domain Sampling
dtc=pi/(2*alpha*Tx);    % Compressed Time sampling
dx=c/(4*b0);            % Range Resolution

% Signal characteristics
```

```
Ts=(2*(xc-x0))/c;
Tf=(2*(xc+x0))/c+Tp;       %%%%

% Sampling characteristics
n=2*ceil((.5*(Tf-Ts))/dt);
t=Ts+(0:n-1)*dt;
x=xc+.5*c*dt*(-n/2+1:n/2);
dw=pi2/(n*dt);
w=wc+dw*(-n/2:n/2-1);

end
```

Power Equalization

```
function [ E ] = R_PwrEq( fsb0 )
%PwrEq is a power equalization and windowing function
% Power equalization
%
mag=abs(fsb0);
amp_max=1/sqrt(2);     % Maximum amplitude for equalization
afsb0=abs(fsb0);
P_max=max(afsb0);
I=find(afsb0 > amp_max*P_max);
nI=length(I);
fsb0(I)=((amp_max*(P_max^2)*ones(1,nI))./afsb0(I)).* ...
        exp(1i*angle(fsb0(I)));
%
% Apply a window (e.g., power window) on fsb0 here
%
E=sum(mag.*abs(fsb0));
%

end
```

Target Generation

```
function [ xn, fn ] = R_targets( ident, x0, ntar )
%Targets generates a user defined number of targets at either static
%locations or dynamic
% x0 is the range mid point and ntar is the # of targets to
% be generated
if ident==1
    xn = (-.9 + 1.8.*rand(ntar,1)).*x0;
    fn = .4 + .6.*rand(ntar,1);
elseif ident==0
    xn(1)=-.5*x0;           fn(1)=.75;
    xn(2)=0;                fn(2)=1;
    xn(3)=.5*x0;            fn(3)=.5;
    xn(4)=.65*x0;           fn(4)=.65;
end
end
```

APPENDIX B

The MATLAB code contained in Appendix B is used to generate the figures used to illustrate the results presented in the Cross-Range Imaging section.

Cross-Range Imaging

```
clear all;

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% ASSUMPTIONS: Tx > Tp;                                %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

% Define constants and signal parameters
global NOISE;
NOISE=rand(1,100000);       % r=a+(b-a).*rand  [a,b]
[y,fc]=CR_SigGen([0 0],0,0);
[mc,xc,y0,yc,uc,L,kus,kuc,m,dkuc,dku,u,k,ku] = CR_initialize(fc);

%%%%%%%%%%%%%%%
%  Simulation %
%%%%%%%%%%%%%%%

% Target definitions
ntarget=7;      % used only when ident=1
ident = 0;      % set ident to 1 for random target gen
                % ident=0 is static 4 target generation
if ident==0
    ntarget=4;  % test case for static vs. dynamic target #s
end
[yn,fn]=CR_targets(ident,y0,ntarget);

% initialize reflected signal
s=zeros(1,mc);

% reflected
for i=1:ntarget
    % Target distance calculation along entire synthetic aperture
    dis=sqrt(xc^2+(yc+yn(i)-uc).^2);
    % Signal generation and reflectivity combine for reflected sig
    s=s+fn(i)*CR_SigGen(dis,uc,L);
end

%%%%%%%%%%%%
% Compression-interpolation-decompression to unalias signal %
%%%%%%%%%%%%
% s=s.*exp(1i*kus*uc);       % Orig sig before bb conv for squint
dis0=sqrt(xc^2+(yc-uc).^2); % Reference aperture distance calculation
cs=s.*conj(CR_SigGen(dis0,uc,L)); % Compressed signal
```

```
cs1=[cs,cs(mc:-1:1)];    % Append mirror in slow-time to reduce
                        % errors in interpolation
fcs=fty(cs1);            % FFT of compressed signal WRT u

mz=m-mc;                % # of zeros is 2*mz
fcs1=(m/mc)*[zeros(1,mz),fcs,zeros(1,mz)];  % zero padded compressed
sig

figure(1)
plot(uc,real(cs))
xlabel('Synthetic Aperture u, meters')
ylabel('Real Part')
title('Compressed Aperture Signal')
axis('square')
axis([uc(1) uc(mc) 1.1*min(real(cs)) 1.1*max(real(cs))]);

figure(2)
plot(dkuc*(-mc:mc-1),abs(fcs))
xlabel('Synthetic Aperture Frequency ku, rad/m')
ylabel('Magnitude')
title('Compressed Aperture Signal Spectrum')
axis('square')
axis([-mc*dkuc mc*dkuc -.05*max(abs(fcs)) 1.05*max(abs(fcs))]);

figure(3)
plot(dku*(-m:m-1),abs(fcs1))
xlabel('Synthetic Aperture Frequency ku, rad/m')
ylabel('Magnitude')
title('Compressed Aperture Signal Spectrum after Zero-padding')
axis('square')
axis([-dku*m dku*m -.05*max(abs(fcs1)) 1.05*max(abs(fcs1))]);

cs=ifty(fcs1);
cs=cs(:,1:m);           % Remove mirror image in slow-time

figure(4)
plot(u,real(cs))
xlabel('Synthetic Aperture u, meters')
ylabel('Real Part')
title('Compressed Aperture Signal after Upsampling')
axis([u(1) u(m) 1.1*min(real(cs)) 1.1*max(real(cs))]);
axis('square')

dis=sqrt(xc^2+(yc-u).^2);   % uncompressed reference distance
s=cs.*CR_SigGen(dis,u,L);    % Decomp sig
s=s.*exp(-1i*kus*u);    % Bb conversion for squint

fs=fty(s);

% Reference Signal calculation
kx=4*(k^2)-(ku+kus).^2;
kx=sqrt(kx.*(kx > 0));
fs0=(kx > 0).*exp(1i*kx*xc+1i*(ku+kus)*yc+1i*.25*pi);
```

```
% Slow-time Matched Filtering
F=fs.*fs0;
f=ifty(F);

figure(5)
plot(u,abs(f))         % also try “real” and “imag” parts of “f” array
xlabel(‘Cross-range y, meters’)
ylabel(‘Magnitude’)
title(‘Cross-range Reconstruction’)
axis(‘square’)
axis([-y0 y0 0 1.1*max(abs(f))]);
```

Signal Generation

```
function  [y, fc] = CR_SigGen(dis,SynAptArr,L)
global NOISE;

fc = 300e6;
c=3e8;

n1=floor(length(NOISE)/length(dis));
noise=[];
for i=1:length(dis)
    noise = [noise NOISE(((i-1)*n1)+1)];
end
y=exp(-1i*2*((2*pi*fc)/c)*dis).*(abs(SynAptArr) <= L);
```

Target Generation

```
function [ yn, fn ] = CR_targets( ident, y0, ntar )
%Targets generates a user defined number of targets at either static
%locations or dynamic
% y0 is the cross-range mid point and ntar is the # of targets to
% be generated

if ident==1
    yn = (-.9 + 1.8.*rand(ntar,1)).*y0;
    fn = .4 + .6.*rand(ntar,1);
elseif ident==0
    yn(1)=0;                fn(1)=1;
    yn(2)=.75*y0;           fn(2)=1;
    yn(3)=.5*y0;            fn(3)=1;
    yn(4)=-.75*y0;          fn(4)=1;
end
end
```

Initialization

```
function [mc,xc,y0,yc,uc,L,kus,kuc,m,dkuc,dku,u,k,ku] =
CR_initialize(fc)

c=3e8;
pi2=2*pi;
lambda=c/fc
```

```
k=pi2/lambda;           % Wavenumber

L=400;                  % Synthetic Aperture is 2*L
yc=0;                   % Cross-range distance to CoTA
y0=100;                 % Target area within [yc-y0,yc+y0]
xc=5e2;                % Range to center of target area

theta_c=atan(yc/xc);    % Squint angle to CoTA
Rc=sqrt(xc^2+yc^2);     % Squint range to CoTA
kus=2*k*sin(theta_c);   % Doppler freq shift in ku domain

% Perform slow-time compression to save PRF %
xcc=xc/(cos(theta_c)^2);    % redefine xc for squint processing
                                % ^ unneeded due to broadside
assumption
                                % but left in for possible future work
du=(xcc*lambda)/(4*(y0+L)) % Sample spacing in aperture domain
duc=(xcc*lambda)/(4*y0);    % Sample spacing for compressed aperture
DY=(xcc*lambda)/(4*L)      % Cross-range resolution

L_min=max(y0,L);            % Zero-padded aperture

%
% u domain parameters for compressed signal
%
mc=2*ceil(L_min/duc);       % # of samples on aperture
uc=duc*(-mc/2:mc/2-1);      % Synthetic aperture array
dkuc=pi2/(mc*duc);          % Sample spacing in ku domain
kuc=dkuc*(-mc/2:mc/2-1);    % kuc array
dku=dkuc;                   % Sample spacing in ku temp
%
% u domain parameters and arrays for Synthetic Aperture signal
%
m=2*ceil(pi/(du*dku));      % # of samples on aperture
du=pi2/(m*dku);
u=du*(-m/2:m/2-1);          % Synthetic aperture array
ku=dku*(-m/2:m/2-1);        % ku array

end
```

APPENDIX C

The MATLAB code contained in Appendix C is used to generate the figures used to illustrate the results presented in the Synthetic Aperture Radar section.

SAR Simulation

```
clear all; close all

global NOISE;
NOISE=randn(1,100000);
c=3e8;
pi2=2*pi;
f0=200e6;
w0=pi2*f0;
% sigType = input(‘2 = AWGN, 1 = LFM Chirp, 0 = Rect Pulse: ‘);
% 1 for LFM Chirp, 0 for Rect Pulse
sigType = 2 ;
[y,wc,wcm,Tp,alpha]=SAR_SigGen2(0,w0,sigType);
colormap(gray(256))

% f0=50e6;                      % Bb BW is 2*f0
% w0=pi2*f0;
fc=wc/pi2;                     % Carrier freq
% wc=pi2*fc;
lam_min=c/(fc+f0);            % Wavelength min/max
lam_max=c/(fc-f0);            %
kc=(pi2*fc)/c;                %
kmin=(pi2*(fc-f0))/c;         % Wavenumbers, min,max,center
kmax=(pi2*(fc+f0))/c;         %

xc=500;                       % Distance to CoTA
x0=200;                       % [xc-x0, xc+x0]
y0=400;                       % [-y0,y0]

D=2*lam_max;                  % Diameter of planar radar

Bmin=(xc-x0)*tan(asin(lam_min/D));  % Min 1/2 beamwidth
Bmax=(xc+x0)*tan(asin(lam_max/D));  % Max 1/2 beamwidth
L=Bmax+y0;                    % Synthetic aperture length is 2*L

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%   u domain parameters for SAR signal    %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
du=D/4;                       % Sample spacing in u domain
du=du/1.2;                    % 10% guard band
m=2*ceil(L/du);               % # of samples on aperture
dku=pi2/(m*du);               % Sample spacing in ku domain
u=du*(-m/2:m/2-1);            % Synthetic aperture array
ku=dku*(-m/2:m/2-1);          % ku array
```

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%   u domain params compressed for SAR    %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
x_t=[xc-x0:2*x0:xc+x0];
lam_t=[lam_min:lam_max-lam_min:lam_max];
B_t=x_t(:)*tan(asin(lam_t/D));
L_t=B_t+y0;
duc=min(min((x_t(:)*lam_t)./(4*L_t)));

clear x_t lam_t
dkuc=dku;
mc=2*ceil(pi/(duc*dkuc));   % #of samples on aperture
mc=2^ceil(log(mc)/log(2));  % chosen to be a power of 2
duc=pi2/(mc*dkuc);
uc=duc*(-mc/2:mc/2-1);      % SynAp array
dkuc=pi2/(mc*duc);          % spacing in ku domain
kuc=dkuc*(-mc/2:mc/2-1);    % kuc array

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%  Fast-time domain parameters and array  %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
Ts=(2*(xc-x0))/c;              % Start time
Tf=(2*(xc+x0))/c+Tp;           % end time
T=Tf-Ts;                    % Fast time interval
Ts=Ts-.1*T;                 % Guard band
Tf=Tf+.1*T;                 % "
T=Tf-Ts;
theta_ax=asin(lam_min/D);
Tmin=max(T,(4*x0)/(c*cos(theta_ax)));

dt=1/(4*f0);                % time sampling
n=2*ceil((.5*(Tf-Ts))/dt);     % # of time samples
t=Ts+(0:n-1)*dt;            % time array
dw=pi2/(n*dt);              % freq sampling
w=wc+dw*(-n/2:n/2-1);       % freq array
k=w/c;                      % wavenumber array


Ik=find(k >= kmin & k <= kmax);
[temp,nk]=size(Ik);
lambda=zeros(1,n);
lambda(Ik)=(pi2*ones(1,nk))./k(Ik);     % Wavelength array
phi_d=asin(lambda/D);        % Divergence angle

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%  Resolution and Parameters of Targets   %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
dx=c/(4*f0);                % range resolution
dy=D/2;                     % cross range resolution

ntarget=4;

xn(1)=-.5*x0;     yn(1)=.4*y0;        fn(1)=2;
xn(2)=.5*x0;       yn(2)=.5*y0;        fn(2)=1.5;
xn(3)=.4*x0;       yn(3)=-.4*y0;       fn(3)=1;
xn(4)=-.4*x0;      yn(4)=-.5*y0;       fn(4)=2;


%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%                Simulation                %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
s=zeros(n,m);

for i=1:ntarget;
  td=t(:)*ones(1,m)-2*ones(n,1)*sqrt((xc+xn(i))^2+(yn(i)-u).^2)/c;
  sn=fn(i)*SAR_SigGen2(td,w0,sigType).*(ones(n,1)*abs(u) <= L &
t(:)*ones(1,m) < Tf);

  sn=sn.*exp(-1i*wc*t(:)*ones(1,m));    % Fasttime Bb conv

  % calculate (w,u) domain beam pattern
  theta_n=ones(nk,1)*atan((yn(i)-u)/(xc+xn(i)));
  beam_p=zeros(n,m);
  beam_p(Ik,:)=(.5+.5*cos((pi*theta_n)./(phi_d(Ik).'*ones(1,m)))).* ...
      (abs(theta_n) <= phi_d(Ik).'*ones(1,m));

  % incorporate beam pattern
  s=s+iftx(ftx(sn).*beam_p);
end

G=abs(s)';
xg=max(max(G)); ng=min(min(G)); cg=255/(xg-ng);
image(t,u,256-cg*(G-ng));
axis('square');axis('xy')
xlabel('Fast-time t, sec')
ylabel('Synthetic Aperture (Slow-time) U, meters')
title('Measured Stripmap SAR Signal - Gaussian Noise')
pause()

td0=t(:)-2*xc/c;
s0=exp(1i*wcm*td0+1i*alpha*(td0.^2)).*(td0 >= 0 & td0 <= Tp);
s0=s0.*exp(-1i*wc*t(:));        % Bb reference fast-time signal

s=ftx(s).*(conj(ftx(s0))*ones(1,m));    % Fast-time matched filtering

G=abs(iftx(s))';
xg=max(max(G)); ng=min(min(G)); cg=255/(xg-ng);
tm=(2*xc/c)+dt*(-n/2:n/2-1);    % fast-time array after matched filt
image(tm,u,256-cg*(G-ng));
axis('square');axis('xy')
xlabel('Fast-time t, sec')
ylabel('Synthetic Aperture (Slow-time) U, meters')
title('Stripmap SAR Signal after Fast-time Matched Filtering - Gaussian
Noise')
```

SAR Signal Generation

```
function  [y, wc, wcm, Tp, alpha] = SAR_SigGen2(td,w0,sigType)
global NOISE;

fc = 1500e6;            % Define carrier freq
wc = 2*pi*fc;
alpha = 0;              % initialize alpha
wcm = wc;               % Temporary wcm definition
n1=floor(length(NOISE)/length(td)); % Determine the number of samples
needed
noise=[];               % Initialize empty noise vector

% Determine Signal type
if sigType == 2     % Noise input selection
    for i=1:length(td)
        noise = [noise NOISE(((i-1)*n1)+1)];
    end
    [a,b]=size(td);
    % Test for ensuring matching matrix sizes
    if a < b
        noise=ones(a,1)*noise;
    else
        noise=noise’*ones(1,b);
    end
    Tp = .2e-6;     % Pulse duration definition
    ybb = 1*(td>=0 & td<=Tp);   % Create ‘window’
    size(ybb)   % used for testing
    size(noise)
    y = ybb.*noise;     % Apply the window to the noise
    y = y.*exp(1i*wc*td);   % upconversion
elseif sigType == 1     % LFM chirp input selection
    Tp = .2e-6;     % Pulse duration def
    alpha = w0/Tp;  % calculate alpha
    wcm = wc-alpha*Tp;  % calculate the modified carrier
    ybb = 1*(td>=0 & td<=Tp);   % window
    y = ybb.*exp(1i*wcm*td+1i*alpha*(td.^2)); % apply window to chirp
else                    % Rectangular Pulse selection
    Tp = .2e-6;
    ybb = 1*(td>=0 & td<=Tp);
    y = ybb.*exp(1i*wc*td);
end
```

LIST OF REFERENCES

[1] Freeman, Tony. *What is Imaging Radar?* [Online]Jet Propulsion Laboratories, January 26, 1996. [Cited: September 20, 2012.]http://southport.jpl.nasa.gov/desc/imagingradarv3.html.

[2] Oleary, Ellen. SEASAT 1978. *Imaging Radar at Southport.* [Online] Jet Propulsion Laboratories, February 10, 1998. [Cited: September 20, 2012.]http://southport.jpl.nasa.gov/scienceapps/seasat.html.

[3] Goodman, Joseph. *Introduction to Fourier Optics: Third Edition.* Englewood, CO: Roberts & Company Publishers, 2005.

[4] A. W. Love. In memory of Carl A. Wiley. *IEEE Antennas and Propagation Society Newsletter,* pp 17–18, June 1985.

[5] Butler, M.J.A. et al., *The application of remote sensing technology to marine fisheries: an introductory manual.* FAO Fish. Tech. Pap., (295):165, 1988.

[6] Soumehk, Mehrdad. *Synthetic Aperture Radar Signal Processing with MATLAB Algorithms.* New York, NY: Wiley & Sons, Inc., 1999.

[7] D.E. Barrick & C.T. Swift. *The SEASAT Microwave Instruments in Historical Perspective.* IEEE J. Oceanic Eng. OE-5, 74-79, 1980.

[8] I.G. Cumming & F.H. Wong. *Digital Processing of Synthetic Aperture Radar Data: Algorithms and Implementations.* Norwood, MA: Artech House, Inc., 2005.

THIS PAGE INTENTIONALLY LEFT BLANK

INITIAL DISTRIBUTION LIST

1. Defense Technical Information Center
   Ft. Belvoir, Virginia

2. Dudley Knox Library
   Naval Postgraduate School
   Monterey, California